I/O News

Space Glizzard: A C-10 Game

Current Versions of Cromemco Software

A Second Look at CP/M Plus

THE OFFICIAL PUBLICATION OF THE INTERNATIONAL ASSOCIATION OF CROMEMCO USERS

Volume Four, Number Two

Single Copy Price $7.50

Cromemco Introduces UNIX System V Computers at NCC in Las Vegas

The recent National Computer Conference in Las Vegas saw the introduction of over a thousand new computer products and computer developments. None was more dramatic than Cromemco's introduction of two new systems using the UNIX System V operating system.

These new products, the System 100 and System 300 series, feature a set of advanced technology developments that move mainframe and mini features to the micro level.

The UNIX System V software, of course, isn't unique. It's one of the fastest growing areas of microcomputers today (see related story). But, Cromemco's announced products support this industry standard operating system with a very unique package:

- The operating system (UNIX System V) includes Berkeley enhancements that improve the utility of the operating system **without changing the standard.**
- The new systems allow expansion of RAM memory to **16 megabytes.**
- The disk drive capacity is standard at 50 megabytes and expandable to 1200 megabytes using SMD drive.
- Cache memory is used in the high speed disk controller for extremely fast disk access.

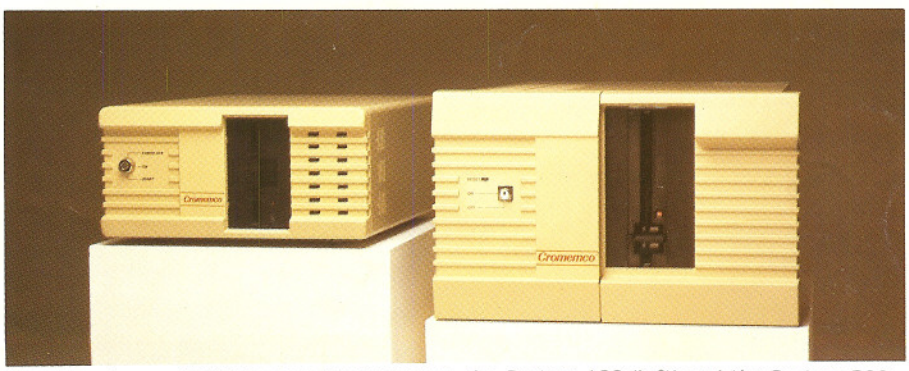
Probably the most general observation that can be made about these new Cromemco systems is that the advanced features that have been designed in are all within the context of industry standards. These standards, such as the UNIX System V operating system, Ethernet networking and communications, the IEEE-696 (S-100) industry-standard bus, the ST-506 interface, and a generic 68000 family 32-bit processor, will enable the user to capitalize on the technical capabilities of the systems while maintaining the ability to take advantage of hardware and software products that use those standards.

Product Details

The new Cromemco systems include two series of machines, each with four, standard configuration models. The System 100, comparable to the Cromemco D-Series System One series, is an 8-slot system and can be expanded to 4M bytes of RAM memory. The number of available slots in a 4 megabyte system configuration allows the support of up to 8 users (up to 16 users can be supported in the 2M byte configuration).

The System 300, which compares to the System Three in the D-Series, is a 20-slot computer for even greater ex-

Continued on page 15



Cromemco's new UNIX System V computers, the System 100 (left) and the System 300.

Cromemco UNIX Q&A

by Dr. Roger Melen

EDITOR'S NOTE:

In response to the flood of questions that arose after Cromemco announced it's entrance into the UNIX world, Dr. Roger Melen, Vice-President and co-founder of Cromemco, Inc., prepared a list of commonly asked questions, and his answers. We will present it serially over the next few issues, beginning with this, Part 1.

1. What is UNIX?

UNIX is a trademark of AT&T Bell Laboratories for a multi-user, multi-tasking operating system written by employees of Bell Laboratories, a research division

Continued on page 8

A Preview of UniPlus+ System V UNIX for Cromemco Systems

by Bill Jaenicke

Lately, everywhere you look you see mention of something called UNIX System V. AT&T is making an all-out effort to make the public aware of what they (and many others) believe to be the great hope for a standard in multi-user, multi-tasking, operating systems. The promise is for software portability: the ability to write programs on one machine and have them run on any other machine that supports the standard operating system. This has long been a dream of programmers and end-users alike.

In keeping with its reputation of being on the forefront of current technologies, Cromemco Inc. announced at the recent NCC their own UNIX offering, UniPlus+ System V, for their new X-series of microcomputers. Prior to this announcement, during the week of June 9th thru June 13th, I had the good

Continued on page 16

ABSOFT FORTRAN 77 FOR CROMEMCO UNIX.

Absoft Fortran 77 Is Currently Supported Under the Following Operating Systems:

- UNOS
- XENIX (V7)
- Uni-Dos
- CP/M 68K
- Regulus
- Vistados
- PDOS**
- AMOS
- VERSAdos
- UNIX V7
- UNIX V7 BSD 4.1
- UNIX III
- UNIX V
- UNIX V/68
- IDRIS
- Apple Workshop**

ABSOFT has been shipping FORTRAN 77 compilers since 1980. They have been used for everything from mapping continents and analyzing seismic data to real time process control and information management by NASA. They are currently installed in hundreds of sites all over the world.

- AVAILABLE NOW
- UNLIMITED PROGRAM AND DATA SIZES
- LINK DYNAMICALLY AT RUNTIME — OVERLAYS
- SOURCE LEVEL DEBUGGER
  - Displays source code concurrent with execution.
  - Allows Single Step, Break points
  - Variable Examination / Assignment, I/O status
  - HARDWARE FLOATING POINT SUPPORT IS STANDARD
- VIRTUAL ARRAY SUPPORT IS STANDARD
  - Allows arrays to exist on disk storage
- SUPPORTS IEEE MATH STANDARDS FOR 32 BIT AND 64 BIT MATH
- EXECUTION SPEEDS IN EXCESS OF 160K WHETSTONES
- STRUCTURED PROGRAMMING EXTENSIONS
  - Select Case Constructs
  - Block Do Constructs
- REQUIRES ONLY 35K OF MEMORY
- ASSEMBLY LANGUAGE SUBROUTINES ARE EASILY INTERFACED
- COMPILER HAS BEEN IN PRODUCTION FOR OVER 4 YEARS
- $995—MANUAL AVAILABLE SEPARATELY—$50

DETAILED SUMMARY SHEET AVAILABLE UPON REQUEST

Royal Data, Inc.
2199 Garden Street
P.O. Box 2745
Titusville, FL 32780

ORDERS AND INFORMATION
(305) 267-1960
TECHNICAL HOTLINE (9:00 to 5:00 EST)
(305) 267-0757

The SIMULATOR with the right combination unlocks the power of Cromix for your CDOS and CP/M programs

Our CDOS-CP/M Simulator is designed to replace Cromemco's simulator in all Z-80 and 68000 Cromix configurations and incorporates powerful features that make your system work more efficiently and effectively.

All CP/M 2.2 Systems Calls are processed and all MP/M system calls are ignored without creating fatal errors.

Dynamic Printer Control — Allows up to 4 printers to operate under program control of the CP/M I/O byte. Printer output can be directed to any device or file you choose. If the selected printer is busy, output will automatically be made to a file that can be printed later. The ^P function of CP/M and ^W/^T functions of CDOS are also supported.

Location of Directories (B: Z:) — can be specified in another environment file (.profile) using **full 128 byte Cromix path names**. A default directory (/lib/ovr) is always searched after the specified directory if a file is not found.

Environment Files — can be created that specify the running environment for specific programs. These files include up to 34 128 byte function keys, printer spooling On/Off, printer selection, upper case lock, circular buffer line count, and CP/M or CDOS emulation flag.

Programmable Function Keys for All Common Terminals — including those without function keys. Once function keys have been programmed, they can be saved to an environment file that is automatically loaded the next time the program is executed.

Printer Spooling — Spooling to the printer device can be enabled with the keyboard or an environment file.

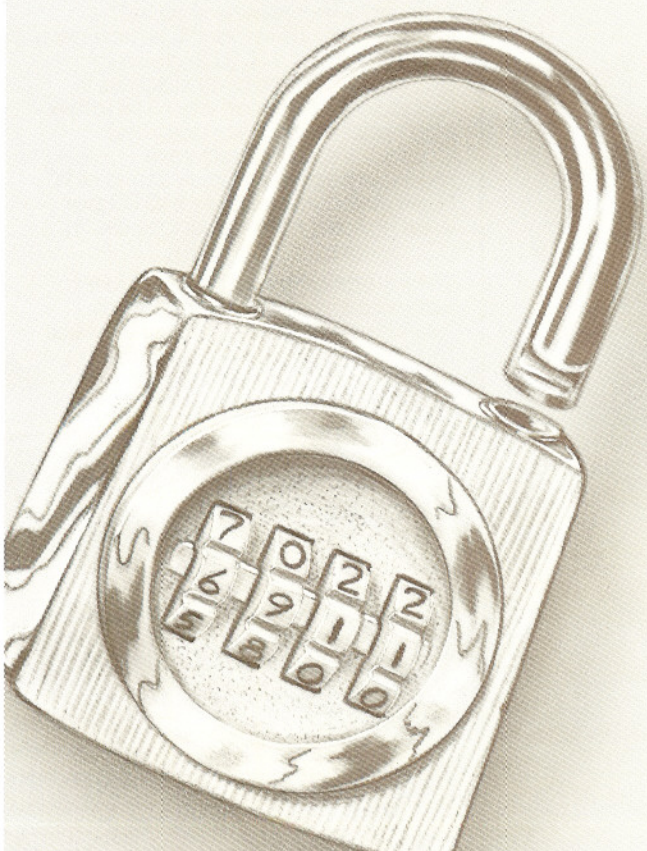
EOF Characters (^Z) — are eliminated from the end of user disk files.

Input Line Editing — As you enter a line of data, the simulator allows interactive editing (exchange, deletion, or insertion) and/or recall of up to 255 lines from the circular line buffer.

Command Function Menu — allows the user to branch to our command menu and selectively change the environment features. In addition the **Fork to a Shell** makes it convenient to leave a program, work on other files and reenter the program where you left it.

Used properly these enhancements will give you greater control and flexibility with your system.

Call for more information — Dealer Inquiries Invited.



CDOS and CROMIX are trademarks of Cromemco, Inc..
CP/M and MP/M are trademarks of Digital Research, Inc.

713/781-6911

Gunn Enterprises, Inc.
5615 Richmond, Suite 150
Houston, TX 77057
Telex: 358282 GEI HOU

MasterCard VISA AMERICAN EXPRESS

table of contents

May/June, 1984 Volume Four, Number Two

I/O News

The Official Publication of The International Association of Comemco Users is available through membership in the association. Editorial and advertising policies are designed for the enlightenment of the members in regard to new uses for, and developments of, Cromemco products and other products compatible with Cromemco systems.

COVER FEATURES

ARTICLES & FEATURES

DEPARTMENTS

I/O News (ISSN 0274-9998) is published bi-monthly by The International Association of Cromemco Users (a California corporation), P.O. Box 17658, Irvine, CA 92713. General offices are at 4750 Von Karman Avenue, Suite 450, Newport Beach, CA 92660. Telephone: (714) 955-0432. Second-class Postage Paid at Santa Ana, CA. **POSTMASTER: Send address changes to I/O News, P.O. Box 17658, Irvine, CA 92713.**

Subscriptions to I/O News are entered with membership in The IACU. Yearly memberships may be purchased for $35 (U.S. delivery address), $41 (delivery address in Canada or Mexico), and $48 (other international delivery address). Contact IACU for multi-year membership rates. Back issues of I/O News are available for $10.00 per issue. Please note: all prices are in U.S. dollars.

Return postage must be included with all manuscripts and photos submitted if they are to be returned. The IACU and I/O News accept no responsibility for the return of unsolicited materials.

Richard Kaye, Editor and Publisher
Lynn L. Platzek, Advertising/Production Manager
William E. Jaenicke, Technical Editor
Art Direction: Howard Millman
Printing: The Dot Generator
Typography: Dynacomp

input...

Editor:

As those of us who use 32K Structured BASIC know, it truly is a very powerful language with many features not found on other BASICs. However, I have come across an occasional, but interesting, quirk. On certain programs, if I erase a **ksam** file and then attempt to recreate it, I receive an "error 169." If I attempt to erase the file again, I get an "error 128." For example:

```
Erase "Testfile.dat"
Kcreate \100,10\"Testfile.dat"
Error 169 — KSAM Primary File Exists
Erase "Testfile.dat"
Error 128 — File Not Found
```

Notice I said "certain programs." This error will not occur in all cases! It also only happens under CROMIX and will not act up as often under sim.bin version 00.34. A "cure" is as follows:

```
Erase: "Testfile.dat"
Kopen \1\"Testfile.dat"
Kadd \1,Key$(-1)\Data$(-1)
Close
Kcreate \100,10\"Testfile.dat"
```

I realize this looks like a gross misprint. However, I have several programs on which this performs correctly!

As a general note, I tried **Dsk "@:"** but it did not do any good either. Until I found this fix, I had to exit BASIC before the file could be recreated.

With best regards,
William J. Williford
Systems Solutions, Inc.
2122 E. Highland Avenue
Suite 420
Phoenix, AZ 85016
(602) 224-0026

Editor:

I enjoy reading I/O News despite the fact that I am neither a programmer nor an electrical engineer, but only a dilettante in Cromemco and dBASE II. There are two features in I/O News which would be extremely helpful to amateurs like myself.

The first is to regularly publish the latest versions of all Cromemco software. One of the really difficult problems with unsophisticated users is appraising whether or not particular software programs will function on Cromemco systems.

This point brings me to the second suggestion. Could you refer me to software produced by third parties which will run "barefoot" under CDOS 2.56? For examples, WordStar or other advanced word processing programs and Supercalc or other spreadsheets.

I noted with interest the letter from George Collier, Jr. [Vol. III, No. 5] where he states he runs Supercalc II, version 1.0, on a Cromemco System Three under CDOS 2.56. On page 15 of the same issue is an ad by Capital Electronics for software configured for the Cromemco, and it mentions WordStar and Supercalc II. Does this mean that these were modified, or are they the same programs one purchases anywhere?

It would be a great help to non-professional users of Cromemco systems to be able to determine—prior to purchase—whether or not certain software will operate effectively with their systems.

As a point of interest, I have a Cromemco Z2D with two 5" DS/DD floppy drives and a 5 Mb hard disk. I use dBASE II most of the time, and find it to be the best database management system I have yet encountered.

I enjoy your many features and commend you on your clear presentations and style.

Sincerely,
Herbert A. Selenkow, M.D.
Brigham Medical Group
454 Brookline Avenue
Boston, MA 02215

Dear Dr. Selenkow:

Your letter raises several points, all of which have application to many members. First, you will note that in this issue we have the current versions of Cromemco software. We will try to make this information part of every issue from now on.

Second, if you have questions about any particular software, you can write or call us and we will look into it for you. That is part of what you get with your IACU Membership. [For the benefit of other readers, we have already responded to Dr. Selenkow about the questions he raised.]

Finally, there has been a project under way within the Product Marketing Department at Cromemco for over two years to research certain third party software and determine which runs "barefoot" on Cromemco systems. That project has resulted in a book which has been sent, or is in the process of being sent, to all dealers. We expect to receive our copy soon, which should speed our response time to members' queries.

Ed.

Dial-up Bulletin Boards for Cromemco Users

More and more interest is growing regarding dial-up bulletin board systems that cater to Cromemco users. Since we became equipped with a modem, our own interest has also grown. Happily, we have learned of two existing bulletin boards that serve those with Cromemco equipment.

The first one is known as the CrosNest, and is located in San Mateo, California. The sysop (BBS lingo for system operator) is Wilbur Smith (IACU member 00607). The BBS is called the DATATECH BBS & RCP/M system and operates under CPM. The system offers a variety of services, among which are messages between users and facilities for the uploading and downloading of programs and files. There are a considerable number of public domain programs available. The system is online 24 hours a day, 7 days a week. There is no initiation or membership fee. You simply dial up (at 300 or 1200 baud) and follow the prompts to get set-up with a user ID. The phone number is: 415 - 341-9336.

output...

Lynn Platzek

The other system is as yet un-named, and is located in Phoenix, Arizona. The sysop is Dan Lepinski (IACU member 00007). The BBS system software was developed in-house at Professional Data Systems, and is written in C to operate under the CROMIX operating system. It can accommodate up to five users simultaneously. The primary purpose of this system is for information interchange between Cromemco users. Messages can be deposited and read within the groupings of Hardware, Software, Technical, and Miscellaneous. In addition there is a mail facility for sending messages to specific "boxes." Although the details have not been worked out, there will be a nominal membership fee to cover operating expenses. For more information contact Mr. Lepinski at 602 - 265-6656.

Both of the system operators of the above mentioned systems have offered to write descriptive articles regarding their BBS, which we will publish when available.

And Now, BBS for CDOS?

While we are on the topic of Bulletin Board Systems, does anyone have information on BBS software that operates under CDOS? If so, Mr. John W. Todd would like to hear from you (and so would we). He can be contacted at:

18 Walnut Road
Lanexa, Virginia 23089
Work #: (804) 865-6162

Inaugural CRAMP Plans Taking Shape

Plans for the first CRAMP [CRomemco cAMP] are being finalized, and reservations are still coming in at a steady pace, according to Richard Quinn, organizer of what is hoped will become an annual event. The idea behind CRAMP is to provide a forum for the sharing of information about Cromemco systems and related products. It is geared more toward "super users" rather than novices, and is attracting a sizable number of suppliers of peripheral products and third party software, as well as dealers and sophisticated users.

The event, taking place over three nights and four days starting at noon on September 11th, will be held at Camp Calamigos Ranch in the Santa Monica Mountains. The site, chosen for its low pressure environment for such high pressure topics, offers a minimum of distractions. Attendees are advised to leave their dressy clothes at home, but bring jeans and flannel shirts, shorts and T-shirts, bathing suits and cowboy hats. This is a summer camp, not a luxury hotel, with the focus of attention on the subject matter. And the subject matter promises to be dynamic.

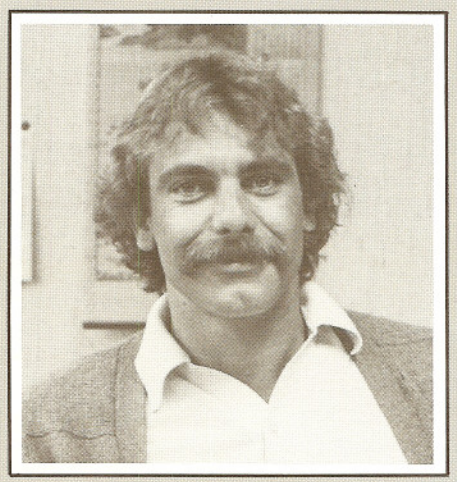
From noon on Tuesday [9/11], until the 6:00 pm California Ranch dinner, conference participants will have an opportunity to install their hardware and software demos, and get acquainted with others as they arrive.

The first session starts after dinner. Called "Patch City," it is an opportunity to learn the software tricks the pros have used for years to create faster versions of WordStar, dBASE II, text editors, and other popular programs. Aside from "fast patches," attendees are invited to demonstrate those clever patches that allow otherwise unworkable programs to run on Cromemco systems. And that's just the first session!

Immediately following this there will be a schedule review [who knows what new products will appear?], UNIX demos, with a California wine tasting fete to round out the evening.

The next morning starts with a "Wrangler Style" [catch your own cow?] breakfast followed by applications software and operating system sessions until the noon break for a real Mexican style lunch. [Tums will be provided free by Richard Quinn.]

The Wednesday afternoon sessions include such topics as networks, modem software, bulletin boards, elec-



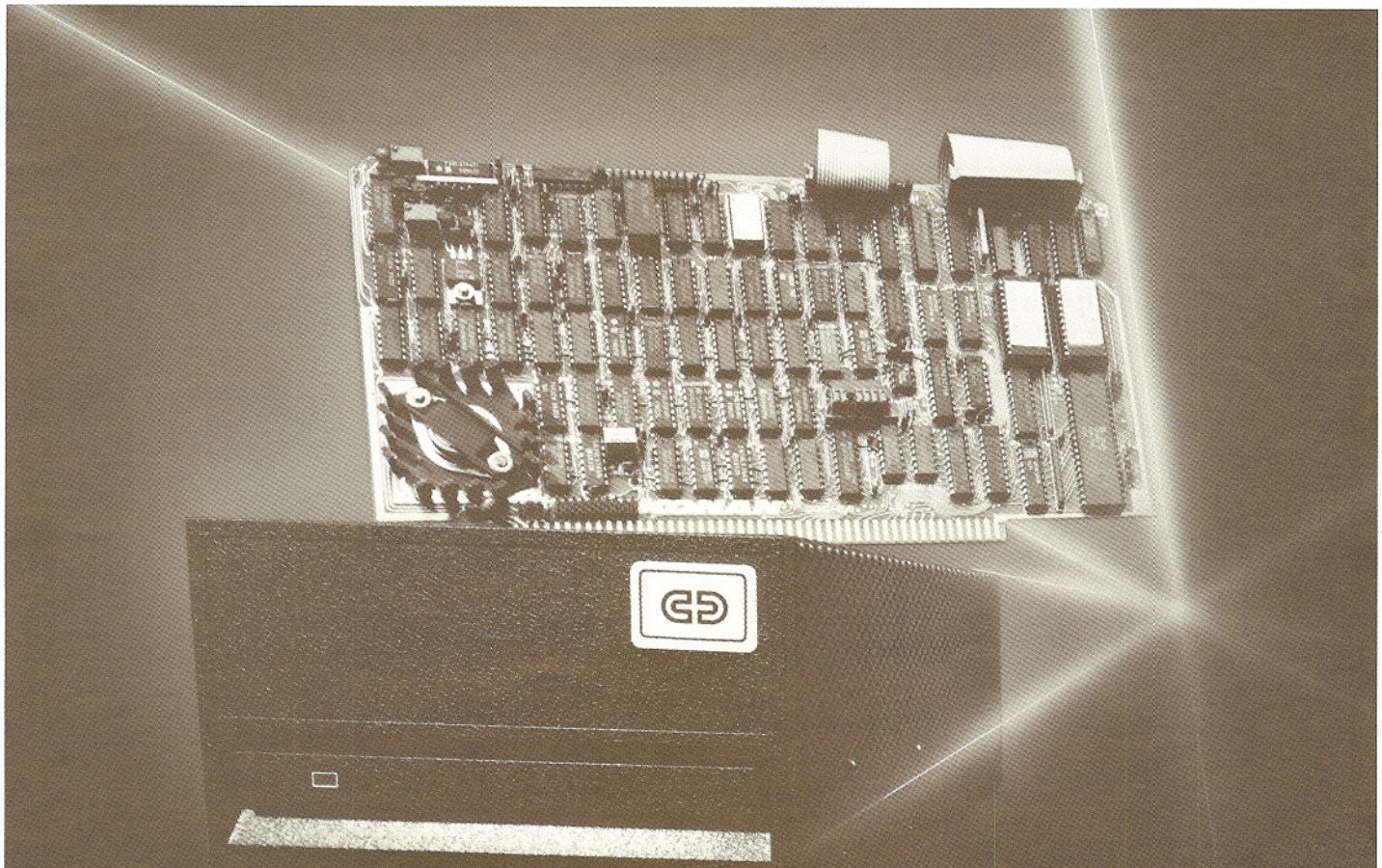
Bill Jaenicke

tronic mail services, hard disk drive demos and benchmarks, and back-up devices and techniques. All-in-all, a busy afternoon. But, after dinner, there's a chance to relax around the ol' campfire with some live [albeit, amateur] entertainment before the open house featuring "super systems" and showing of video tape training aids.

Thursday morning's sessions include new Cromemco offerings [new now, new in the near future, and exploring the company's path to anticipate long-term announcements], followed by a valuable hardware trouble-shooting forum. Like we said, CRAMP is not for novices.

The afternoon is devoted to user group contributions, I/O News feedback and critiques, what Cromemco users can expect from the publishing world and the published word, culminating with computing power enhancers, and future designs using co-processors.

Then comes the fun break, with the balance of the afternoon devoted to such diversions as a volleyball game [Olympic style?], Tug 'o War [will this become the next Olympic event?] and participated in by teams such as the CROMIXs vs. the UNIXs. Another surprise dinner, campfire, and more open system time will spell an end to the day.



The Hard Disk That's Hard To Believe

35 Megabytes/35 Milliseconds

Introducing the Performance Solution for your Cromemco CROMIX System 1,* System 2 or System 3.* 35 Megabytes of hard disk storage with 35 millisecond average access time.

Complete Package Includes:

- Control Data Corporation Hard Disk
- Disk Controller
- System Generation program
- Extensive Disk Diagnostics
- Efficient disk I/O drivers provide additional shell buffers
- Custom CROMIX I/O and Utilities
- One Year Warranty (H.E.L.P. Maintenance Service Available)

*Configurations for CS-0, CS-1, CS-1H, CS-3, and Z-2H vary from $3995. to $4495.

This new 35 Megabyte Hard Disk Subsystem is available only from **Gunn Enterprises, Inc.**

$ 3,995.*

Dealer Inquiries Invited.

Gunn Enterprises, Inc.
5615 Richmond, Suite 150
Houston, TX 77057
713/781-6911 • Telex: 358282 GEI HOU

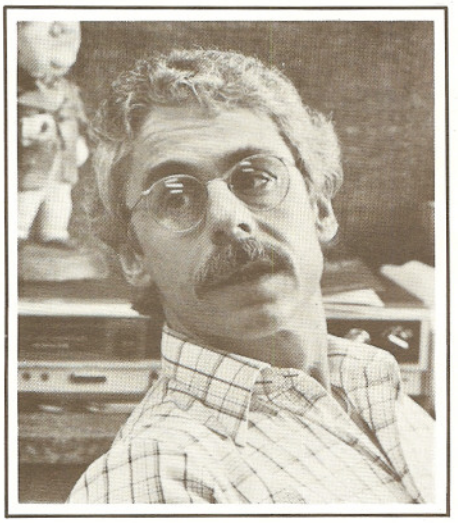
MasterCard VISA AMERICAN EXPRESS

Friday morning's opening forum is listed as "Cromemco Feedback." Would you like to influence future developments, air some gripes, perhaps even compliment the founders [we hope one of them will be there] on what they've done right? This is your chance to get it off your chest.

For those who have to catch an early flight, you'll miss the mid-morning session, an in-depth update on UNIX. Those who stay will enjoy a noon lunch with keynote speaker, and a chance for farewells until the next CRAMP.

Initial reservations indicate that attendees will be arriving from many places. Regular shuttles between LAX [Los Angeles International Airport] and the camp will be run on Tuesday and Saturday. Special trips can be arranged for those with unusual scheduling demands.

We at I/O News applaud Richard Quinn for the efforts he is expending to make the inaugural CRAMP an event worth repeating. For reservations or more explicit information, contact Quinn at: QUINTEC Services, Inc., 30313 Canwood Street, Agoura, CA 91301. Telephone: (818) 889-4819. We hope to see many of you there.



Richard Kaye
Editor

UNIX SYSTEM V Q&A

Continued from front cover

of AT&T (the pioneer of the United States telephone system). Cromemco installed UNIX on the Cromemco X-series of computer hardware (included in the new CS-100 and CS-300 models) under license from AT&T and Unisoft Incorporated. Unisoft Inc. is a company that has assisted Cromemco in the installation of UNIX System V and certain enhancements developed by the University of California at Berkeley. Unisoft Inc. refers to this software combination by their trademark: UniPlus+ System V.

2. What versions of UNIX does Cromemco use?

Cromemco uses the most recent standard version of AT&T UNIX System V, Release 1.0. This is a version currently supported by AT&T, the originator of UNIX.

AT&T has announced a second release of UNIX System V which incorporates some minor utility enhancements. Cromemco will offer this release when it is ready for widespread distribution. Cromemco offers a software update service, SUDS, for Cromemco users so that one can be assured of always having the latest versions of Cromemco software.

3. What is UNIX System III? How is it different from UNIX System V?

UNIX System III was an earlier release of UNIX from AT&T. This earlier release has never received the official support of AT&T. UNIX System V offers higher performance than System III.

There are other versions of UNIX from AT&T. The earliest widely used UNIX is called VERSION 6. This version was disseminated in the late 1960's. It was followed by VERSION 7 which was introduced about ten years ago. System III followed VERSION 7. Most recently, System V succeeded System III and is the current version of UNIX distributed by AT&T. Thus, there have been four major versions of UNIX distributed by AT&T which have been made available through greater than fifteen years of development. With the most recent version, System V, AT&T has officially stated their support of the product.

4. I've heard a lot about Berkeley UNIX 4.2. What is it and why didn't Cromemco use it?

Like all popular software, UNIX has a large and active user's group. The University of California is one such group and has played an important role in the evolution of UNIX. However the only version sold and supported by AT&T is UNIX System V. Cromemco has added various utility programs from U.S. Berkeley to UNIX System V (thus the references in the literature of "Berkely enhanced"). These programs and drivers are only added when they do not affect the standardization of UNIX System V. Some of the Berkeley enhancements do affect the standardization, and those are not included in the Cromemco software.

Dr. Roger Melen, Vice President and co-founder of Cromemco, Inc. One of his favorite indoor pastimes is answering questions on UNIX.

5. I've heard a lot about XENIX. What is it, and how is it different from UNIX System V?

XENIX is a product of Microsoft Inc., Seattle, Washington based upon an AT&T license of UNIX. This software is supported by Microsoft for a number of computers. The current version is based upon AT&T UNIX System III.

Microsoft says they will introduce a version of XENIX based on UNIX System V later this year.

6. What are some of the features that might attract me to the Cromemco UNIX computers over some of the others?

The CROMEMCO UNIX computers have been designed and manufactured to be number one in quality, performance, and implementation of industry standard software and interfaces. There are specific features which exemplify these goals. Namely, in comparison with others the Cromemco systems offer:

a. More RAM memory: practical configurations with physical memory up to 16 Megabytes (only 8 memory boards,

Fast Tape Backup...

immediate delivery

FAST TAPE BACKUP

Pacific Datanet's 5¼" tape streamer backs up at a consistent fast 5Mb per minute, using standard 90 ips, ¼" tapes.

VARIETY OF CAPACITIES

Only *Pacific Datanet* offers 5¼" winchesters in so many capacities — 12/11Mb, 19/17Mb, 26/21Mb, 40/32Mb, 50/43Mb, 70/62Mb, 105/92Mb, and 140/125Mb.

SOFTWARE

Pacific Datanet's menu-driven software makes installation simple.

Pacific Datanet's flexible segmentation scheme allows large capacity drives to function using normal CROMIX utilities.

COMPLETE SYSTEMS

Pacific Datanet's subsystems are complete. Package includes a 5¼" winchester, our controller, host adaptor, heavy duty 120/240 power supply, cables and software.

ERROR CORRECTION

Pacific Datanet makes CRC obsolete by providing real error correction.

RELIABILITY

Pacific Datanet's stringent quality control assures reliable operation.

PRICE

Prices start at $2,195 for the hard disk subsystem and $1,995 for the streaming tape.

For further information contact the Pacific Datanet Dealer near you *or* call

Pacific Datanet

PACIFIC DATANET, LTD.
4701 Patrick Henry Drive, Bldg. 9
Santa Clara, CA 95054
(408) 980-0693
Telex: 759341

leaving seven free slots in a CS-300 with 50Mb hard disk).

b. More Disk memory: High capacity 5 inch 50Mb hard disk and SMD interface for up to 1200Mb total storage.

c. Demand paging memory manager hardware: Single instruction context switch for rapid process changing. Some of the others use segmentation type memory managers which load down heavily in rapid context switch applications.

d. Standard (non-proprietary) processor: Standard 68000 family (10 MHZ) processor; software portability.

e. Floating Point IEEE hardware math support.

f. Many Industrial and Scientific interface board options.

g. Standard Ethernet with TCP/IP protocol used in ARPANET and Defense Data Network.

h. Multiprocessor I/O architecture for faster I/O speed.

i. Cache memory hard disk interface for fast disk memory access.

j. Multi-language support: Fortran 77, C, Pascal, COBOL, BASIC.

k. Database support: Informix and Unify.

l. Error-correcting RAM memory (1 bit correction, 2 bit detection).

m. Maximizer (TM) 12 Mips (million instructions per second) co-processor Program accelerator for high performance number-crunching. Operates in CROMIX partition of UNIX-based system.

n. Cartridge tape disk backup device: Economical and convenient 24-megabyte unit.

o. 9 track 1600 BPI Standard Tape drive: Provides interchange with other systems.

p. World-wide service and support.

7. How much RAM memory does UNIX require on a Cromemco system to get started?

Cromemco recommends that a minimum of 512K of RAM memory be used for all UNIX systems. The memory requirements of the basic operating system are difficult to estimate because UNIX freely exchanges programs and parts of the operating system between RAM and the disk memory system. In a 512K system programs of over 300K may be executed. Of course more memory is required to run programs larger than 300K.

8. How much RAM memory does UNIX take for multi-user applications? (or, What is Swap Space anyway?)

UNIX provides more than 300K of user space in a 512K RAM system. A surprising feature of UNIX is that all users of a multi-user system may concurrently use all of the 300K file. This is because UNIX maintains an area on the hard disk memory system called the Swap Space. This Swap Space area typically occupies four megabytes or more of space on the disk.

The size of the available Swap Space is important. Say five users are concurrently running different programs, each approximately 300K in size. This 300K use is comprised of approximately 270K of program plus 30K of buffers allocated by the operating system to the user. They attempt this on a 512K RAM machine. On other operating systems the users would receive "out of memory" error messages, except for the first user who would be able to execute his program without problem.

UNIX, on the other hand, loads programs from the normal hard disk file system into RAM memory and then back to Swap Space if there is no RAM available at the time of user execution initiation. This allows all users to share the 300K RAM on a time slice basis. Each program runs slower because of the sharing. During the times when the programs are not running in RAM they are temporarily moved to Swap Space to store them while the other programs are running. It is necessary that the Swap Space be big enough or the system will give an error.

Five users of 300K each (approximately 270K of program plus 30K buffers) require more than 1200K Swap Space, since four of the users would be swapped out at any given time. While swapping does allow sharing, many users find the reduction in throughput during swapping a serious limitation. Therefore, for five happy 300K users a better implementation would be to have 1500K of user space or approximately 1700K (1.7 Meg) including the OS. If the system is not used by five simultaneous 300K users very much, less memory may be used (with a minimum of 512K).

Fortunately, Cromemco systems are fast at swapping, in comparison with others. This is due to the high performance of the cache memory STDC disk controller.

9. Isn't swapping the same as virtual memory?

Yes and no. Swapping technically uses the disk memory as a low speed, but economical, substitute for RAM memory. This is what virtual memory does, too. But what people really are looking to do when they ask for a virtual memory computer is run a single program that is larger than the RAM user space in the computer. If we take for an example a computer with 300K of user space, we may like to run a 900K program on it without program changes such as "chaining" or "overlays."

"Roger, how about telling me again what that little dooleywocker on the edge does."

A virtual memory operating system on a computer with a memory manager capable of virtual operation will allow the user to run a program up to a maximum size, commonly 8 megabytes, by using 8 megabytes of the disk memory system as a supplement to the available user RAM. The user RAM is typically much smaller than 8 megabytes (1 megabyte might be typical).

Cromemco computers have 16 megabytes RAM capability, twice that of many virtual memory computers. A sixteen megabyte of physical RAM computer costs somewhere around $85,000. Many other computers with 8 megabytes virtual memory capabilities are limited in the physical memory you can have (VAX 780 maximum physical memory is 6.5 megabytes, Apollo DN440 maximum physical memory is 3.5 megabytes). They have to have virtual memory because they do not have the large physical memory technology. These computers can cost much more

than $85,000 and run large programs slower than the Cromemco computer because of the speed of the physical memory.

Virtual memory systems run slower than non-virtual memory systems. Virtual (disk) memory is 1000 times or more slower in random access speed than RAM memory. How much slower the total machine goes depends upon how much you use virtual memory. The more you use it the slower it goes because the more frequent the exchanging of program betwen RAM and disk.

Virtual memory was important when RAM was $50,000 per megabyte (only six years ago). Competition and microcomputer technology have brought the price of RAM below $5,000 per megabyte for single unit buyers, and below that for volume quantity buyers.

Cromemco will have virtual memory when AT&T makes it available for UNIX System V. Currently only non-standard, user-group-modified versions of UNIX have virtual memory. Such a case is the University of California at Berkeley modified UNIX called version BSD 4.2. This version is not supported by AT&T. The AT&T version of UNIX V with virtual memory is scheduled to be available in early 1985.

10. What is demand paging? Isn't that swapping?

In any kind of swapping technique, that is for both the user-space-limited swapping or the virtual-address-space swapping methods discussed in the two previous questions, the operating system decides how much to swap if swapping is to be done.

There are two common amounts of programs swapped from RAM to the disk by different operating systems: 1) big pieces of programs, or 2) little pieces of programs. The swapping of big pieces occurs when entire user programs or major portions of user programs are swapped. The swapping of little pieces occurs when there is sufficient memory management hardware sophistication and accompanying software to allow little pieces of programs to be swapped. The little pieces of user programs swapped are called pages. When little pieces are swapped it is called paging. Demand paging is simply a method by which paging is used.

Cromemco supports demand paging in hardware with the XMM memory management board. For the Cromemco XMM these "little" pages are 4096 bytes. This minimum swap entity, the 4096 byte page, is swapped in response to operating system demand. Swapping is to and from the RAM and the disk. These pages are also used for other functions, such as the minimum size RAM that can be reassigned to a logical address. There is no limitation imposed by this page technique on program address space. For instance in Z-80 based systems the 64K bank limited the user space for a single program to 64K. In the XMM there are 4096 pages of 4096 bytes which support the full 16 megabyte capability of the 68000. Thus a programmer treats paging as if it did not exist and presumes a single 16 megabyte linear address space.

Demand paging is an operating system method by which it: 1) copies pages to memory from disk on an as-needed basis to support the execution of the current task (virtual memory) and 2) manages pages of RAM required by tasks other than the current task. This second issue is important to maximizing system performance. If a large running task of several megabytes is succeeded in the next time slice by a small few kilobyte task, this small task may swap code from the hard disk into a few pages of RAM occupied by the much larger task. When the execution is returned to the large task it is desirable that the few pages of RAM disrupted by the swapping are all that are required to be restored before execution of the larger task. This second benefit offered by demand paging operating systems can result in a significant performance difference. UNIX System V currently supports the second benefit of demand paging system. The XMM provides the support required for this benefit as well as the future virtual memory extensions which are anticipated to be added to future releases of UNIX.

11. Can I run CROMIX software on the CS-100 or the CS-300?

Yes. CROMIX (20 series) is included with all Cromemco UNIX configurations. During boot up, CROMIX actually boots first and then loads UNIX. This makes diagnostics easier since much less of the hardware has to work for CROMIX to run than for UNIX to run. Thus, you can run CROMIX and UNIX on the CS-100 and CS-300. They share the same hard disk, but not the same file structure. You may boot up from one to the other from the hard disk, but multi-users must log off. You may exchange data between file structures using the TAR backup software.

12. Can I run the CROMIX CDOS and third-party CP/M emulators under UNIX?

No. UNIX System V is kept standard. However, CROMIX is supplied standard with UNIX systems, sharing the same hard disk. These routines may be under CROMIX.

13. How much disk memory is used by the operating system and utilities on CS-100 and CS-300 UNIX-based computers?

UNIX and utilities require twenty-one megabytes if you include the Programmer's Software Tools, the Documenter's Software Tools optional packages, disk swap space, and CROMIX. CROMIX occupies approximately one megabyte of this space and the swap space is initialized to approximately four and one-half megabytes.

It is possible to operate with as little as eight megabytes if the on-line manual is removed and the optional packages are not included.

14. Why is so much disk space required for UNIX?

UNIX uses more memory than some other systems because of the reserved swap space, the large size of the on-line manuals, the elaborate time-accounting

Continued on page 23

	AT&T					CROMEMCO	
	AT&T MICRO/SUPERMICRO LINE					Series 100	Series 300
	3B2/300	3B2/300	3B5/100	3B5/200	3B20S	CS-100H50X5	CS-300H50X20E
CPU	WE32000	WE32000	WE32000	WE32000	AMD2901	68000	68000
CPU Performance			.63MIPS	.8MIPS	1 MIPS	1 MIPS	1 MIPS
CPU Clock Rate	7MHz	7MHz	10MHz	10MHz		10MHz	10MHz
OP. System	UNIX V	UNIX V	UNIX V	UNIX V	UNIX V	UNIX V	UNIX V
Berkeley Enhancements	No	No	No	No	No	Yes	Yes
Minimum Memory	512Kb	1Mb	1Mb	2Mb	2Mb	512Kb	2Mb
Maximum Memory	2Mb	2Mb	8Mb	8Mb	12Mb	4Mb	16Mb
ECC Memory	No	No	Yes	Yes	Yes	Optional	Yes
Floppy Disk	2-360Kb	2-360Kb	No	No	No	1-390Kb	1-1.2Mb
Hard Disk-Std.	10Mb	32Mb	48Mb	48Mb	300Mb	50Mb	50Mb
Hard Disk-Optional	No	No	160Mb	160Mb	675Mb	Yes-SMD	Yes-SMD
Tape Back Up	No	No	Yes	Yes	Yes	Yes	Yes
Cartridge Tape	No	No	No	No	No	Yes	Yes
Max. No. Users	18	18	40	60	100	16	16
Software	C Fortran UUCP Workbench Series 3BNet	C Fortran UUCP Workbench Series 3BNet	C Fortran UUCP Workbench Series 3BNet	C Fortran UUCP Workbench Series 3BNet	C Fortran77 Ratfor Cobol Basic Workbench Series 3BNet	Fortran77 C Pascal Basic Cobol Animator Forms Gen. Informix Unify PST DST BNet UUCP	Fortran77 C Pascal Basic Cobol Animator Forms Gen. Informix Unify PST DST BNet UUCP
Price	$ 9,950	$ 15,500	$ 57,000	$ 73,000	$230,000	$ 9,995	$ 19,995

A SECOND LOOK AT CP/M Plus

by Dr. Jerome Freedman and
Dr. R. David Freedman

Introduction

Since we released CP/M Plus for Cromemco computers in November, 1983, several articles have discussed the performance and features of CP/M Plus. To our knowledge, none of the articles has taken an approach similar to ours: banking out disk drivers as well as directory and data buffers; nor has any mentioned software implementation of interbank memory moves. By implementing the concepts developed in this article, we were able to increase the TPA (Transient Program Area) for user programs from 54k to 60k, with or without our hard disk support modules, with no apparent degradation in performance.

Background

CP/M Plus was designed for Z80-, 8080-, and 8085-based micro computers to take advantage of memory that is capable of being separated into many segments—or memory banks, as they are called—in order to extend the amount of memory available to the operating system*. (CP/M Plus is also able to execute in a non-banked environment, but the loss of TPA compared to CP/M version 2.2 makes it of little practical value.) The memory configuration in a banked system can be contrasted to the memory configuration in a non-banked system as follows: a non-banked system can utilize only 64k of continuous memory; however, in a banked system there is a band of memory at the high end of the memory spectrum common to all the banks, and blocks of memory in 2 to 16 independent banks, each of which can be as large as 60k.

BIOS Organization

The CP/M Plus Operating system consists of a hardware independent component, called the BDOS, and a hardware dependent component called the BIOS. In a banked environment, there are two parts to both the BDOS and the BIOS. Portions of the BDOS and the BIOS (called RESBDOS and RESBIOS) are resident at the top of the common memory. The banked portion of the BIOS (called BNKBIOS) sits above the banked portion of the BDOS (called BNKBDOS) in bank 0, or what is called the system bank. The system bank as well as additional memory banks are used to hold data buffers and other information required by the BDOS or the BIOS.

The advantage of a banked system of CP/M Plus is a high performance file system that operates at full efficiency in a hardware environment with bank-switching capabilities. The additional memory is used to provide for increased directory and data buffer space. Because record buffers are maintained in memory on a least-recently used scheme and directory information can be directly accessed through hashing techniques, access time to information on disks is significantly reduced.

At MICAH, because we were configuring CP/M Plus for Cromemco computers that have some—but not all—of the hardware features necessary, we had to devise software routines to do some of the work that Digital Research envisioned as hardware dependent. In particular, we have software routines that move data from place to place within memory banks, and even **between** memory banks. The major work is done by the routines ?XMOVE, ?MOVE, and BNKINF that will be described below.

Having also fully implemented the TYPE byte within the extended disk parameter header (XDPH), we have been able to provide a BIOS that (1) automatically adjusts to single- or double-sided diskettes in either single or double density both 5¼" and 8" formats; (2) supports hard disks; (3) takes full advantage of the increased power of a system with banked memory; and (4) provides the user with 60K of TPA.

The large TPA is achieved by "banking off" most of the BIOS into bank 0, the system bank. The following is maintained in common memory:[1]

(1) **BOOT.ASM[2]:** The parts of the cold- and warm-boot routines that load the CCP (?LDCCP, ?RLCCP, and the routines that use the BDOS).

(2) The disk parameter blocks (DPB's) for each of the possible drive types, including the hard disk(s).

(3) RWBUF, a 512-byte buffer used for data transfer.

(4) **MOVE.Z80[3]:** the entire memory management module.

(5) All character I/O drivers,[4] except the initialization routine.[5]

(6) **BIOSKRNL.ASM[6]:** the DRI supplied root module of the relocatable BIOS, as described in the DRI System Guide.

The rest of the BIOS is in the system bank.

Implementation of Banked Disk Drivers

We now come to the technical discussion of what has been referred to as the banking out of the disk drivers for CP/M Plus. Most of the concepts developed in our November, 1983 release of SUPER BIOS PLUS were carried over from previous developments in our product line with one major exception. The floppy disk drivers were reorganized substantially to take advantage of the data structures provided by CP/M Plus. Consequently, we were able to reduce the overhead in reading the disk label to determine the type of media which was present.

The concept for banking out the disk drivers was developed in our MP/M product, and an excerpt of the listing of our BNKXIOS is shown in Listing 1. Here we see that all disk access functions are routed through a single entry point labelled XDISK (line 362). For example, the home drive function is implemented at the label XHOME on line 255, which simply places the value of the home function, HOMEF, in register H and jumps directly to XDISK for execution.

The interesting thing to note in this listing is the operation of the XREAD (lines 267ff.) and XWRITE (lines 315ff.) functions. XREAD calls XDISK to read the requested sector and then calls the BANKMOV routine to transfer the data from the buffer in common memory to the address requested by the user unless the disk buffer happens to be the directory buffer, in which case no move has to take place. The BANKMOV routine first switches into the user's bank by calling SWTUSER, and concludes by switching back to the system bank by calling SWTSYS.

In the case of a write to disk, the process is reversed. First, the user's bank is selected and the data moved into the common buffer, if necessary; and after a switch back to the system bank, the data is finally written to disk.

```
0026  ; EXTERNAL ROUTINES USED BY XIOS
0027  ;
. . . . . .
0035   GLOBAL NULLDEV
. . . . . .
0056  ;
0057  ; EXTERNAL ROUTINES USED BY XIOS
0058  ;
. . . . . .
0060   GLOBAL DISPOLL  ;POLL DISPATCHING ROUTINE
. . . . . .
0064  ; OTHER ENTRY POINTS AND REFERENCED MODULES
0065  ;
. . . . . .
0070   GLOBAL DIRBF  ;DIRECTORY BUFFER
0075  TRUE: EQU 0FFFFH       ;VALUE OF "TRUE"
0076  FALSE: EQU NOT TRUE
0077  ;
0078  ;
0079  ; DISK SYSTEM EQUATES
0080  ;
0081  MAXFL: EQU 4  ;MAXIMUM NUMBER OF
                                        ;FLOPPIES
0082  MAXHD: EQU 1  ;MAXIMUM NUMBER OF
                                        ;HARD DISKS
0083  SELDF: EQU 0  ;SELECT DISK ROUTINE
                                        ;CALLED
0084  HOMEF: EQU 1  ;HOME ROUTINE CALLED
0085  TRKF: EQU 2  ;SELECT TRACK ROUTINE
0086  SECF: EQU 3  ;SET SECTOR
0087  READF: EQU 4  ;READ ROUTINE CALLED
0088  WRITEF: EQU 5  ;WRITE ROUTINE CALLED
0089  TRANF: EQU 6  ;TRANSLATE FUNCTION
0090  DMAF: EQU 7  ;SET DISK BUFFER
                                        ;FUNCTION
0091  ;
0092  ; FLOPPY DISK FUNCTIONS
0093  ;
```

```
0094    GLOBAL SELDSK  ;SELECT DISK
0095    GLOBAL HOME    ;HOME DRIVE
0096    GLOBAL SETTRK  ;SET TRACK
0097    GLOBAL SETSEC  ;SET SECTOR
0098    GLOBAL READ    ;READ SECTOR
0099    GLOBAL WRITE   ;WRITE SECTOR
0100    GLOBAL SECTRAN  ;TRANSLATE SECTOR
0101    GLOBAL SETDMA  ;SET DMA BUFFER
. . . . . .
0255  XHOME:
0256  ;
0257  ;************************************************
0258  ;
0259  ; XHOME ROUTINE HOMES THE DRIVE AFTER CALLING
0260  ; XDISK TO PREVENT ANY INTERRUPTS FROM OCCURRING.
0261  ;
0262  ;************************************************
0263  ;
0264   LD H,HOMEF
0265   JR XDISK
0266 ;
0267  XREAD:
0268  ;
0269  ;************************************************
0270  ;
0271  ; XREAD ROUTINE READS THE REQUESTED SECTOR AFTER
0272  ; CALLING XDISK TO PREVENT ANY INTERRUPTS FROM
0273  ; OCCURRING.
0274  ;
0275  ;************************************************
0276  ;
0277   IF MAXHD GT 0
0278   LD A,READF
0279   LD (RWOPR),A ;SET UP FOR READ
0280   LD H,A
0281   CALL XDISK  ;DI AND PERFORM READ
0282  ;
0283  ;THIS ROUTINE TRANSFERS DATA FROM THE SELECTED BANK
0284  ;TO THE COMMON BUFFER.
0285  ;
0286  BANKMOV:
0287   PUSH AF  ;SAVE RETURN CODE
0288   CALL SWTUSER  ;SWITCH TO USER'S BANK
0289  ; DI
0290   LD DE,BUFF  ;GET USER'S DISK BUFFER
0291  DSKBF: EQU $-2
0292   LD HL,DIRBF
0293   AND A
0294   SBC HL,DE
0295   JR Z,MOVOK
0296   LD HL,COMBUF ;GET INTERNAL BUFFER
0297   LD BC,80H
0298   LD A,0 ;GET READ/WRITE OPERATION
0299  RWOPR: EQU $-1
0300   CP READF
0301   JR Z,BNKOK
0302   EX DE,HL
0303  BNKOK:
0304   LDIR
0305  MOVOK:
0306   CALL SWTSYS  ;SWITCH BACK TO SYSTEM
0307  ; DI
0308   POP AF
0309   RET
0310   ELSE
0311   LD H,READF
0312   JR XDISK
0313   ENDIF
0314  ;
0315  XWRITE:
0316  ;
0317  ;************************************************
0318  ;
0319  ; XWRITE ROUTINE WRITES THE REQUESTED SECTOR AFTER
0320  ; CALLING XDISK TO PREVENT ANY INTERRUPTS FROM
0321  ; OCCURRING.
0322  ;
0323  ;************************************************
0324  ;
0325   IF MAXHD GT 0
0326  ; DI
0327   LD A,WRITEF
0328   LD (RWOPR),A
0329   CALL BANKMOV
0330   LD H,A
0331   JR XDISK
0332  ;
0333  XSETTRK:
0334  ;
0335  ;************************************************
0336  ;
0337  ; XSELTRK SIMPLY DISPATCH TO CORRECT CONTROLLER
0338  ;
0339  ;************************************************
0340   LD H,TRKF
0341   JR XDISK
0342  ;
0343  XSETSEC:
0344  ;
0345  ;************************************************
0346  ;
0347  ; XSELSEC SIMPLY DISPATCH TO CORRECT CONTROLLER
0348  ;
0349  ;************************************************
0350   LD H,SECF
0351   JR DISPDISK
0352  ;
0353  XSECTRAN:
0354  ;
0355  ;************************************************
0356  ;
0357  ; XSECTRAN SIMPLY DISPATCH TO CORRECT CONTROLLER
0358  ;
0359  ;************************************************
0360   LD H,TRANF
0361  ;
0362  XDISK:
0363  ;
0364  ;************************************************
0365  ;
0366  ; XDISK DISABLES ANY TIMER INTERRUPTS FROM
0367  ; INTERFERING WITH DISK ACCESS ROUTINES.
0368  ; IT ALSO SWITCHES IN BANK ZERO FOR HARD DISK
0369  ; VERSIONS.
0370  ;
0371  ;************************************************
0372  ;
0373  ; DI
0374  ;
0375  DISPDISK:
0376  ;
0377  ;************************************************
0378  ;
0379  ; DISPDISK SELECTS THE PROPER CONTROLLER AND
0380  ; DISPATCHS TO THE REQUESTED ROUTINE.
0381  ;
0382  ;************************************************
0383  ;
0384   LD A,0  ;CONTROLLER NUMBER
                              ;FILLED IN ABOVE
0385  CONTR: EQU $-1
0386   OR A,A
0387   LD A,H
0388   JR Z,DIS010
0389   ADD A,7
0390  ; DI
0391  DIS010:
0392   CALL DISPOLL
0393  ;
0394  ; FLOPPY DISK FUNCTIONS
0395  ;
0396  FDRTNS:
0397   DW SELDSK  ;SELECT DISK
0398   DW HOME  ;HOME DRIVE
0399   DW SETTRK  ;SET TRACK
0400   DW SETSEC  ;SET SECTOR
0401   DW READ  ;READ SECTOR
0402   DW WRITE  ;WRITE SECTOR
0403   DW SECTRAN  ;TRANSLATE SECTOR
0404   IF MAXHD GT 0
0405  ;
0406  ; HARD DISK FUNCTIONS
0407  ;
0408  HDRTNS:
0409   DW NULLDEV  ;SELECT DISK
0410   DW NULLDEV  ;HOME DRIVE
0411   DW NULLDEV  ;SET TRACK
0412   DW NULLDEV  ;SET SECTOR
0413   DW NULLDEV  ;READ SECTOR
0414   DW NULLDEV  ;WRITE SECTOR
0415   DW NULLDEV  ;TRANSLATE SECTOR
0416   ENDIF
0417  ;
0418  XSETDMA:
0419  ;
0420  ;************************************************
0421  ;
0422  ; XSETDMA SIMPLY SETS THE INTERNAL DISK BUFFER
0423  ;
0424  ;************************************************
0425   LD (DSKBF),BC
0426   LD A,B
0427   AND A,C
0428   INC A
0429   JP Z,SETD  ;FOR CLEARING BUFFERS
0430   LD HL,DIRBF
0431   AND A
0432   SBC HL,BC
0433   JR Z,SETD
0434   LD BC,COMBUF ;MUST BE COMMON BUFFER
0435  SETD:
0436   CALL SETDMA  ;FOR FLOPPIES
0437   JP NULLDEV  ;FOR HARD DISK
0438  HDDMA: EQU $-2 ;ADDRESS FILLED IN WHEN
                              ;HARD DISK IS PRESENT
```

This concept was implemented somewhat differently in the CP/M Plus BIOS because CP/M Plus requires that on entry to the read or write routine, the address of the extended disk parameter header (XDPH) be in the <DE> register pair, and that the parameters for the read or write operation be contained in the public variables @ADRV, @RDRV, @TRK, @SECT, @DMA, and @DBNK. The TYPE byte, which is utilized by BIOS for media determination, is located at XDPH – 1. Also, all operations refer to an internal stack called BIOSST (line 000).

As can be seen from Listing 2, an excerpt from our 16FDC module, what we do is to load register <A> with the read/write flag, <BC> with the size of the disk buffer necessary for the density of the source (or destination) diskette, and <HL> with a pointer to the appropriate routine, READ or WRITE. Registers <DE> have on exit, as they had on entry, a pointer to the address of the XDPH.

```
0126  readf: equ 1 ;disk read
0127  writef: equ 0 ;disk write
. . . . . .
0205  fdread:
0206   ld hl,read          ;address of read:
0207   ld a,readf
0208   jr rwcomon
0209  fdwrite:
0210   ld hl,write         ;address of write:
0211   ld a,writef
0212  rwcomon:
0213   push af  ;save operation
0214   ld bc,512  ;default to ddens
0215   dec de  ;point to type byte
0216   ld a,(de)  ;get it
0217   inc de  ;restore proper address
0218   and a,40h  ;check density
0219   jr nz,isdd          ;if double density
0220   ld bc,128  ;single density
0221  isdd:
0222   pop af              ;restore operation flag
0223   jp bnkinf  ;in MOVE.Z80
0224  ;
```

When we arrive at the bank-in function, we will save the operation flag and the address of the appropriate routine

(READ or WRITE) pointed to by <HL>; and we will switch to a local stack. Next, we will check to see if we are trying to do the disk I/O from or to common memory or bank 0, where direct-DMA I/O is possible. If we are in common or bank 0, then we simply call the READ or WRITE routine, restore the stack set on entry, and return to the calling routine.

On the other hand, if we are not reading to common memory or bank 0, we will exchange the address of our read-and-write buffer (RWBUF) in common memory for the DMA address requested by the calling program. After reading the data to RWBUF, we will use BANKMOV (lines 217-239) to move the data to the requested location, and then we will reselect bank 0, where the calling BDOS routine is located.

Similarly, if we are writing to disk from anywhere but common memory or bank 0, we will use BANKMOV to move the data from the requested location to RWBUF, reset the DMA address to be that of RWBUF, and then we call the write routine. The details are given in Listing 3.

```
0142  bnkinf:
0143   ld (rwoper),a ;save read/write function
0144   ld (rwaddr),hl ;save read/write rtn address
0145   ld (bdosst),sp ;save bdos stack
0146   ld sp,biosst ;set up bios stack
0147   push ix  ;save user's ix
0148   push de  ;transfer xdph
0149   pop ix  ;to ix for bios read/write
0150
0151  ;
0152  ; determine data bank
0153  ;
0154   ld a,(@dbnk)       ;bank requesting read/write
0155   or a,a  ;in bank zero?
0156   jr z,nobank ;yes, don't need bankinf
0157
0158  ;
0159  ; determine dma address
0160  ;
0161   ld de,(@dma)
0162   ld a,d  ;get high order addr of dma
0163   ld hl,combas ;point to common data page
0164   cp a,(hl)  ;above common?
0165   jr nc,nobank ;yes, don't need bankinf
0166
0167   push de  ;save bdos' dma
0168   ld hl,rwbuf ;point to internal buffer
0169   ld (@dma),hl ;set the new dma
0170
0171  ;
0172  ; now we are ready to do the actual read or write
0173  ;
0174  dorw:
0175   ld a,readf
0176  rwoper: equ $-1
0177   cp a,readf          ;if not read, do a write
0178   jr nz,dowrite
0179   push bc              ;save the environment
0180   push de
0181   push hl
0182   push ix
0183   pop de
0184   call rwfunc          ;do the read or write
0185   pop hl              ;restore the enviroment
0186   pop de
0187   pop bc
0188   push af              ;save error code
0189   call bankmov         ;move data to requested dma
0190   pop af              ;restore error code
0191   jr rwret            ;and return
0192
0193  dowrite:
0194   ex de,hl            ;move data to rwbuf
0195   call bankmov
0196   push ix
0197   pop de
0198   call rwfunc          ;write from rwbuf
0199
0200  rwret:
0201   pop de  ;recover old dma
0202   ld (@dma),de        ;restore it to @dma
0203   jr ixret            ;and return
0204  ;
0205  ; arrive here when bankinf not needed
0206  ;
0207  nobank:
0208   call rwfunc  ;do the read/write function
0209  ixret:
0210   pop ix  ;restore user's ix
0211   ld sp,(bdosst)      ;restore stack pointer
0212   ret
0213
0214  ;
0215  ; Support routines for bankinf
0216  ;
0217  bankmov:
0218   ld a,(@dbnk) ;get data bank
0219   call ?bank  ;select it
0220   ldir                ;move the data
0221   sub a,a  ;set back to bank 0
0222
0223  ;
```

```
0224  ; bank select routine
0225  ;
0226  ; input: select bank in register a
0227  ;
0228  ?bank:
0229   push bc  ; save register b for temp
0230   ld b,a  ; bank to b
0231   inc b  ; for looping
0232   sub a,a  ; clear a register
0233   scf
0234  bankloop:
0235   rla   ; rotate left
0236   djnz bankloop
0237   out bankp,a  ; select bank
0238   pop bc  ; restore b
0239   ret
0240
0241  ;
0242  ; dispatch to read/write function
0243  ;
0244  rwfunc: jp 0 ; to the address
                                    ; that was saved above
0245  rwaddr: equ $-2
```

Implementation of XMOVE

One of the memory moves to be optionally supported by the BIOS is an interbank memory move, say for example, 128 bytes from location 1024 of bank 2 to location 256 of bank 1. The routines used are ?XMOVE and ?MOVE. The sample MOVE.ASM provided with CP/M Plus has the code shown in Listing 4 for ?XMOVE and ?MOVE:[7]

The comment in line 11 that Altos cannot perform interbank moves (meaning, "Altos cannot perform direct interbank transfers as a **hardware** function") is the reason that ?XMOVE only contains the RET instruction. The GENCPM utility looks at the first instruction in ?XMOVE; and if it finds a RET instruction, it considers interbank moves to be impossible. GENCPM will therefore put all the directory and data buffers into common memory,[8] with a resultant loss of TPA space. However, if there is another instruction first, GENCPM knows that ?XMOVE has been implemented and that ?MOVE can be used for interbank transfers. Directory and data buffers can consequently be located in the system bank or in the additional banks. Since the ?XMOVE routine in the sample BIOS for Altos computers is a null routine, the sample ?MOVE routine is used only for intrabank transfers.

The importance of our software implementation of ?XMOVE and ?MOVE is that—contrary to the implications of the Digital Research documentation—we can bank off almost all of the disk drivers, and can therefore increase the user TPA.

The ?MOVE routine needs to know whether ?XMOVE has been called since the last time that ?MOVE was called. We provide this information by initializing the source bank storage byte SBANK to −1, and by setting it back to −1 at the end of the ?MOVE routine (lines 113-114). Any subsequent call to ?XMOVE will overwrite the −1 with the source bank for the next call to ?MOVE, and since the −1 will not be found, ?MOVE will know it has to make an interbank move. Because ?XMOVE can be used by ?MOVE only one time,[9] and because the first call to ?MOVE after a call to ?XMOVE will not exceed 128 bytes[10] our 512-byte RWBUF is adequate.

The relevant code can be found in Listing 5, an excerpt from our MOVE.-Z80 module, lines 83 through 125. The ?XMOVE function simply saves the source and destination banks, so that the next call to ?MOVE can determine if it is an intrabank move or an interbank move. In the first case, the code in lines 119-123 to perform a regular Z80 block move instruction is the same as in the ?MOVE routine of the sample BIOS in Listing 4.

However, in the case of an interbank move, there is a different procedure. After initiating the internal stack, the source bank is selected, and the data is moved into the common memory buffer, RWBUF. Then, the destination bank is switched in and the data is moved from the common memory buffer to the destination address. In the meantime, the stack is utilized to insure that the returned values in the DE and HL register pairs are as specified. Finally, the stack is restored.

```
11  ?xmove:          ;ALTOS can't perform interbank moves
12      ret
13
14  ?move:
15      xchg            ;we are passed source in DE
                        ; and destination in HL
16      ldir            ;use Z80 block move instruction
17      DB   0EDH,0B0H
18      xchg            ;need next addresses in same regs
19      ret
```

```
0083  ?xmove:
0084   ld (sbank),bc
0085   ret
0086
0087  ?move:
0088   ld a,(sbank) ; is it an interbank move?
0089   cp a,-1  ; if -1 then normal move
0090   jr z,regmove
0091  ;
0092  ; arrive here for interbank move
0093  ;
0094   ld (bdosst),sp
0095   ld sp,biosst
0096   call ?bank  ; select source bank
0097   push hl  ; save destination
0098   ex de,hl
0099   ld de,rwbuf ; point to common memory
0100   push de  ; save for later
0101   push bc  ; save count
0102   ldir
0103   pop bc  ; recover count
0104   pop de  ; source for second move
0105   ex (sp),hl  ; dest to hl, end of
                                ; source to top of stack
0106   ld a,(dbank) ; select dest bank
0107   call ?bank
0108   ex de,hl
0109   ldir                    ; move the data
0110   pop hl
0111   ex de,hl            ; restore the next locations
0112   ld sp,(bdosst)      ; and the stack
0113   ld a,-1             ; set the no ?XMOVE flag
0114   ld (sbank),a
0115   ret
0116  ;
0117  ; arrive here for a regular intrabank move
0118  ;
0119  regmove:
0120   ex de,hl  ; we are passed source in
                                ; DE and dest in HL
0121   ldir   ; use Z80 block move instr.
0122   ex de,hl  ; need next addr. in same regs
0123   ret
. . . . . .
0246
0247  ;
0248  ; data area for source and destination banks
0249  ;
0250  sbank: db -1       ; source bank, if -1 then
                                ; no interbank move
0251  dbank: db 0 ; destination bank
0252
0253   end
```

Conclusion

We have been running this system for quite some time now, and have found tremendous performance enhancements in many programs, due to the many data buffers allowed by our XMOVE implementation and the increased TPA due to our BNKINF implementation.

Footnotes

*Theoretically, it would be possible to make this additional RAM available for actual program execution or run-time data storage if the application program were specifically tailored to do so. As a result, all the programs written for 16-bit computers could be implemented for CP/M Plus.

Alternatively, by checking console input for a specific code and pushing the entire microprocessor environment onto a local stack and then switching banks, the BIOS could use the additional available memory for windows.

[1]Cf. CP/M Plus (CP/M Version 3) Operating System System Guide (Pacific Grove, CA: Digital Research, 1982) (Hereafter: Guide), section 3.5, pp. 67-68.

[2]Guide, Appendix I.1, p. 137

[3]Cf. Guide, Appendix I.5, p. 152

[4]Cf. CHARIO.ASM, Guide, Appendix I.2, pp. 140-142

[5]Ibid., lines 27-57

[6]Guide, Appendix E, pp. 117-127

[7]Guide, Appendix I.5, p. 152

[8]Guide, pp. 66-67

[9]Guide, p. 86, Table 4-12

[10]Guide, p. 67

UNIX System V

Continued from front cover

tension of capabilities.

All of the systems feature a new CPU/memory control board combination, the XPU central processor and the XMM Memory Management Unit. The XPU board uses a high performance, 10 MHz 68000-family processor and connects to the XMM through a top edge connector. The XMM board provides demand-paging memory management and scatter loading to increase processing speed and optimize memory utilization. Context switching is done in a single instruction for fast response when switching from user to user, or task to task. Therefore, performance degradation when adding users is minimized.

Other system features that are common to all configurations are a 50 megabyte high speed hard disk drive, driven by Cromemco's new STDC controller with cache memory (previously reported in I/O News, Volume IV, Number 1). The STDC is a high performance disk controller that includes a dedicated microprocessor and 4 tracks of cache memory for high speed operation. It automatically verifies information written to the disk with a read-after-write cycle to assure data integrity. A 64FDC board is included for control of a single on-board floppy disk drive (5-inch, 390K byte in the System 100, and 8-inch, 1.2 megabyte in the System 300).

The difference between various models of the new systems comes in the internal memory of the machine. Cromemco is offering models with 512K bytes, 1 megabyte, or 2 megabytes of RAM in each of the new series. Error check and correction is a standard feature at each RAM level, although a non-ECC model is available at 512K bytes.

In another new move, Cromemco includes both the new UNIX System V operating system and the CROMIX operating system in the purchase price for all models. The operating systems are installed on the hard disk at the factory, so no installation procedure is required by the user.

The UNIX System V operating system is the standard for the new generation of high-performance microcomputers. Cromemco has included the highly-acclaimed Berkeley enhancements, including CSHELL, TERMCAP, UUCP, and VI.

To support UNIX System V, Cromemco will offer two other packages that certain users will find useful. The Programmer's Software Tools package (Model UPST-X) includes important tools, such as a C compiler and the YACC and LEX compiler generators, for the system programmer. Also included in this package is the UNIX ADB debugger.

The Documenter's Software Tools (Model UDST-X) includes software for documentation preparation, including the well-known UNIX programs, NROFF for formatting text to line printers, and TROFF for formatting and transmitting text to typesetting equipment.

System Expansion

With the standard models, the System 100 has one or two open slots for additional system expansion and configuration, and the System 300 has 13 or 14 open slots. These, of course, provide many options for tailoring the systems to user need, using Cromemco's plug-in boards. These options include graphics, communications, memory expansion, multiple users, computation acceleration, networking, instrumentation, control, and magnetic storage expansion.

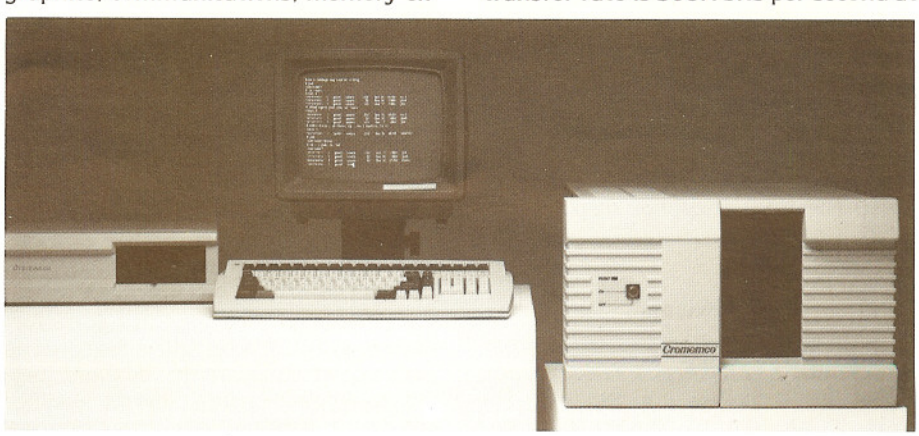
The internal memory (RAM) of the machines can be expanded by adding memory boards. A new 2 megabyte RAM board (Model 2048MSU) has been introduced that provides even more expansion. In fact, the System 100 can be expanded to 4 megabytes and the System 300 to a whopping **16 megabytes** using the new boards.

Systems without ECC memory can be expanded in memory capacity in 256Kb increments by plugging additional 256KZ cards into the system. Systems with ECC can be expanded in either 512Kb or 2048Kb increments by plugging additional 512MSU or 2048MSU cards into the system.

The number of users can be expanded up to 16 by adding OCTART boards, which will accommodate a combination of up to eight users or serial devices per board. (Use one board for eight users, two boards for 16 users). Because of the limited number of board slots available, the System 100 can be expanded to 8 users when 4 megabytes of RAM are included. With 2 megabytes of RAM, fewer boards are used for memory, so the System 100 can accommodate up to 16 users. The System 300, of course, can be expanded up to 16 users and 16 megabytes of RAM.

Hard disk storage can be expanded up to 1200 megabytes by adding the SMDI controller board and interfacing to standard SMD disk drives, which are available from several vendors.

Cromemco is also introducing a new cartridge tape drive for its systems, so adequate tape backup is an easy expansion. The new drive (Model CTD) connects to the 64FDC interface card and, therefore, requires no additional board slots. It has an unformatted capacity of 32 megabytes, which is equivalent to over 85 small (5¼-inch) floppy diskettes and is designed to use standard 1/4-inch tape cartridges. The data transfer rate is 500K bits per second at



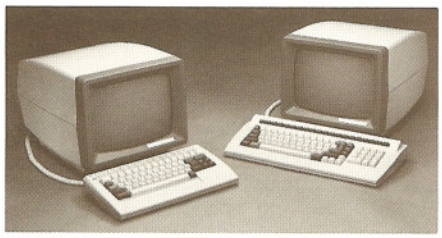
Cromemco's new System 300 20-slot computer.

78 ips.

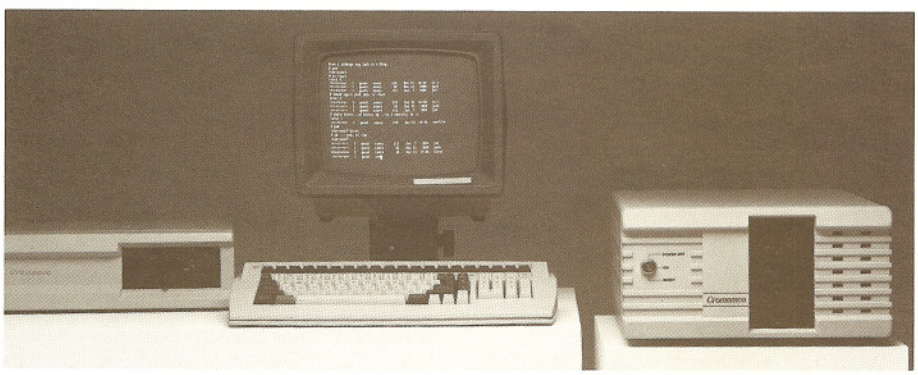
Reel-to-reel tape storage can be added by using the TDS tape drive, which includes the IOP and CSP boards required for operation. Therefore, connecting the TDS to a system requires 2 board slots.

To round out the system, the new System 100 and System 300 computers can use Cromemco's new Model C-5 terminals. They are available with either Cromemco's Model CKBA or CKBC keyboards (the C-5 model numbers are C-5A and C-5C, respectively).

The C-5 terminals have several special features. They can store the contents of up to six screens in memory simultaneously, thereby allowing the user to review the contents of one screen while data is being received in another. In addition, the C-5 has a design feature that increases the lifetime of the display. The terminal will automatically clear the screen, without losing the information, if the terminal is not used after a user-definable period of time. The contents of the screen can be redisplayed by a single keystroke. The C-5 terminals use the P-31 phosphor CRT as standard, with the P-39 phosphor display available as an option.



Cromemco's new C-5 intelligent terminals are available with the CKBA keyboard (left) or the CKBC keyboard, which includes a full numeric pad and 20 function keys.



The new System 100, shown with the C-5C keyboard and the CTD cartridge tape backup drive.

Software

In addition to the operating system, Cromemco has several software packages now available, and more on the way. In addition, of course, there is a large library of UNIX System V software available from other sources.

The software packages that are available immediately for the UNIX System V systems include the following:

PART NO.	DESCRIPTION
CCC-X	C Compiler
FOR-X	Fortran
PAS-X	Pascal
BAS-X	Basic
COB-X	Cobol
ANI-X	Animator (for Cobol)
FM2-X	Forms 2 (for Cobol)
INFX-X	Informix Relational Database
INFXRT-X	Run-Time Informix
CISAM-X	Indexed Sequential Access
B-NET	UNIX networking software

The INFORMIX relational database management system, announced in May for the D-Series, is available for UNIX as part number INFX-X. In addition, Run-Time Informix and CISAM are available for UNIX systems. Part numbers are INFXRT-X AND CISAM-X, respectively.

B-NET is the networking software for the UNIX System V operating system. It implements all seven layers of the ISO model for network protocols. B-NET allows a number of different functions, including interprocess communications between cooperating processes running in machines connected to the B-NET network, remote system access, and implementation of unique end-to-end protocols.

A new board set (Model ENET) is required to implement system networking. The ENET boards (2-card set) make up a high performance controller option that links the Ethernet LAN to the System 100 and System 300 computers. The B-NET networking software, available from Cromemco, implements the standard UNIX protocol for Ethernet (the same protocol used in ARPANET and DDN). The ENET interface requires two system slots.

All in all, the System 100 and System 300 series offer excellent capabilities. The focus on fast operating performance, with such technical features as high speed subsystems, demand paging, re-entrant coding, scatter loading, and cache memory, and the preservation of standards, gives the user state-of-the-art performance with a high degree of flexibility. With the rapid growth predicted for UNIX System V machines, Cromemco remains in the forefront of performance trends, while protecting its customers with paths to upgrade systems performance.

UniPlus+ System V UNIX

Continued from front cover

fortune of attending a Product Training Seminar, held for the benefit of the Technical Support Managers of the Cromemco Regional Offices, at the main Cromemco facility in Mountain View, California. Although many of the latest products were discussed, the main emphasis was on UniPlus+ System V.

My role was one of observer, participant in beta-testing prior to announcement, and most importantly, as reporter. During the course of the seminar a great deal of information was given, and many (but not all) questions answered. Since the seminar, many of the unknown details have been worked out, and will be included as part of this article (readers are also referred to the UNIX Q&A presented in this issue).

HISTORY

Before getting into the specifics regarding UniPlus+ System V, a little history about the evolution of UNIX is in order. UNIX was developed at AT&T Bell Laboratories in 1969, primarily thru the efforts of Ken Thompson and Dennis Ritchie. The thrust of this project was to design an environment in which system programmers could work both effectively and efficiently. Thus, from the beginning, UNIX was meant as a programmer's tool for building tools (programs). It became highly popular among the employees of Bell Labs, growing more elaborate as user requests were implemented.

Before long, a number of educational institutions adopted UNIX for use in their own Computer Science departments. By 1981 there were approximately 1700 installations of UNIX in universities around the nation. Of these, the University of California at Berkeley played a leading role in further development, contributing many additional utility programs and enhancements such as the Programmers Work Bench (PWB). In 1981, a version known as UNIX System III, containing the Berkeley enhancements became available. It, however, was not supported by AT&T.

More recently, AT&T announced its standard UNIX system, which is known as UNIX System V. UniPlus+ System V is Unisoft Inc.'s implementation of

AT&T UNIX System V, specifically ported for Cromemco X-series microcomputers. Today, UNIX represents the largest set of program development and text processing tools of any operating system. But please note, the goal has always focused on programmer productivity: as such UNIX is not what one would call "user-friendly."

THE PRODUCT TRAINING SEMINAR

Three of the five days at the Product Training Seminar were devoted to UniPlus+ System V. The method of instruction was novel: no instructor was present, rather, the "lessons" were presented by what is called "audio-digital courseware" (a trademark of User Training Corporation, the designers of the equipment and course materials). This consisted of a graduated series of tutorial manuals used in conjunction with a specialized cassette playback unit which provided both audio and digital information concurrently, as well as a direct interface to the UNIX system. The audio portion (voice and keyboard sounds) was heard over lightweight headphones; the digital information was displayed on the CRT. So you get the best of both visual and audio assistance in tackling the tutorials. It's like looking over the shoulder of your instructor: you hear what he's saying and see what he types, and then see the results as if the commands had been entered online.

A nice advantage to this method of instruction was that it was self-pacing. Each lesson was outlined, and associated with each topic was a time reference indicating where on the tape the topic could be located. The playback unit had controls for fast-forward and reverse, so you could skip or review topics at will. Most importantly, there was an "online" button which enabled you to interrupt the taped lesson and gain immediate access to the UNIX command line prompt. There you could try out the commands being discussed, and do the experimenting necessary to reinforce that particular subject.

I found this method of instruction to be very effective, as did most of the others in attendance. Complaints were minimal, and concerned such things as the pace being too slow, or that the lecturer's voice was monotonous. The subject matter of the lessons was thorough, and I found it to be a painless way to learn. I understand that this courseware may be made available to dealers to provide instruction to interested users; so long as an agreeable pricing schedule for the lease of the equipment can be worked out (apparently it is quite expensive).

CROMIX and UNIX

In discussing UNIX, a point of reference would be helpful. To contrast UNIX and CDOS would be difficult, for there are virtually no similarities. CROMIX, on the other hand, does provide an excellent point of reference. Not only are there many similarities, but there exists a substantial base of CROMIX users, and they are the prime candidates for making the transition to UNIX.

CROMIX has always been referred to as a UNIX-like operating system. Having now experienced UNIX, I can testify to that statement. I also have a much greater appreciation of CROMIX, for it represents the essence of the UNIX system.

There are many similarities. Both operating systems are multi-user and multi-tasking. They both employ a hierarchical file system in which the disk space is broken into a tree structure of directories and files. The terminology is the same: there is a root directory from which emanates all subordinate directories. There are absolute and relative pathnames associated with any file. Both operating systems have a command line interpreter called the "shell". Many of the utility programs are functionally similar, although their names differ. For example, in CROMIX to determine the current directory you type "d"; in UNIX the corresponding command is "pwd" (for Print Working Directory). Likewise, the CROMIX command to type a file, "ty", is present in UNIX as the command "cat" (for conCATenate—it can also be used to join separate files

into a single file).

The similarities do not end here. As with CROMIX, UNIX also provides features for redirection of I/O (changing the standard input, standard output, or both). It provides for both sequential and detached (background) processing of tasks, and for pipes and filters (linking the output of one process to the input of another). Command scripts can be entered into regular text files by way of a text editor and executed by the shell (in CROMIX these are the familiar "command files"). Password security is provided (the familiar LOGIN and PASSWORD) as well as file access privileges. Lastly, UNIX recognizes "special" files, i.e., device files that have a major and minor number.

Before going into those features more or less unique to UNIX, there are features present in CROMIX which are not accounted for in UNIX. First, CROMIX is faster than UNIX. This is because CROMIX was designed to run in an environment where RAM is plentiful, and disk swapping unnecessary. Second, CROMIX supports record level locking, which is a handy feature in a multi-user environment; UNIX does not. Third, UNIX does not support any form of CP/M simulator (SIM.BIN in CROMIX). This means, quite simply, that none of the applications written for a CP/M (or CDOS) operating system will run under UNIX. At first glance this appears to be a serious shortcoming, but when viewed in light of the amount of software written specifically for UNIX, and taking into account that the Cromemco UNIX systems will come with CROMIX installed, this should not be too great a concern.

UNIPLUS[5] SYSTEM V

As has been noted, there are many similarities between CROMIX and UNIX. Because of this, the transition between using CROMIX and using UNIX should not be too difficult. Where the difference lies is in the sheer number of additional utilities provided by the UniPlus+ System V operating system, and in some additional flexibility that has been built into the shell program. What you have is a much larger tool box, with a more complete array of general and specific utilities. Naturally, it takes some time and practice to become knowledgeable about, and proficient in their use.

UniPlus+ System V contains the majority of the standard UNIX V system utilities. There are additional utilities implemented by UniSoft Inc., many of which perform functions included in the Berkeley UNIX 4.2. Everything covered by a utility in CROMIX has its counterpart in a UniPlus+ System V utility. In addition to numerous other utilities of the same vein, there exists an entire set of system utilization accounting utilities, and numerous text editing programs. Other than the screen-oriented text editor, VI, we didn't have much opportunity to experiment with them. Cromemco definitely had a better idea when they developed the SCREEN Text Editor. Even though VI is screen-oriented, I found it to be considerably more difficult to learn and use than SCREEN. The line-oriented editors, ED and EX, take even longer to get a handle on. This is not to say that they don't do the job well, they do: it's just that learning to use them is not as straightforward.

Because UNIX was designed primarily for system programmers, there is a large set of programming and debugging tools. A 'C' compiler is available (the Cromemco 'C' compiler can be purchased separately) as well as a debugger (ADB). A special utility, LINT, is useful for "cleaning up" 'C' programs before compilation. Other utility programs such as YACC (Yet Another Compiler-Compiler) and LEX (lexical analyzer) can be of assistance if your application requires that you develop a specialized language (a custom database storage/retrieval system, for example).

Bell Labs no doubt generates reams of voluminous documents. Consequently, it is not surprising that UNIX has complete facilities for document pre-

MenuWare™ Believes the Difference Between Success and Failure can be the Quality and Quantity of Information at Your Fingertips.

Document Library is that Difference.

What Is It?

The Document Library is a centralized filing system that allows users to create, store, and retrieve documents in a logical and straight-forward manner. It elegantly bridges the gap between the hardware and the non-technical user while allowing sensible access to information. Your existing word processor files can be easily incorporated into the library. The Document Library solves problems.

Is it Hard To Learn?

Easy to understand menus allow users to operate the Document Library with single-keystroke commands. The user need not know anything about the operating system.

The Library User is assigned a "home group," such as "Accounting" or "Sales" and is given access to as many other groups as necessary. The user can then Create and Find their own documents. He or she can also Review (view, copy, and print) any documents in any group they belong to, but can only Modify (edit, rename, and delete) the documents he or she owns. As an additional feature, a document can be placed under "public" ownership which permits all group members to Modify it. Most Library Users can get "up to speed" without even consulting a manual.

How Do You Find Things?

Documents can be easily retrieved through a variety of avenues: name, owner, group, date, month, year, and keyword in either name or description.

What Does Sensible Access Mean?

Let's face it, the modern electronic workplace has made information management *more* rather than less complicated. Important files, such as costly legal or contractual "boilerplates," can be unwittingly damaged. Sensitive personnel files should not be open for casual inspection.

The MenuWare solution is Sensible Access. The Document Library allows the System Manager to initiate or revise the User Security system at any time. The System Manager has passwords and 25 different groups, or levels of security available. He or she can generate Group Managers and a hierarchical office structure that allows access to files on a need-to-know basis.

Is It Expensive?

The surprising answer is no! Not for a full-featured filing system that will handle over 65,000 documents. The Document Library is a fully self-contained module of the MenuWare Document Processing System. It is compatible with most of the word processing programs using the Cromemco's operating systems. The Document Library represents MenuWare's efforts to bring you the highest return on your capital investment in office equipment available today...real value.

The complete package is only $249.00. When ordering, please specify 5" or 8" disk, for CDOS, CROMIX or C-10 CDOS.

Satisfaction Guaranteed.

Your satisfaction is guaranteed and the call is toll free, inside the continental United States.

To place your order or receive additional information, call (800) 851-MENU or write

MenuWare 16351 Skyline Boulevard Woodside, CA 94062

CDOS, CROMIX and C-10 CDOS are trademarks of Cromemco, Inc.

paration. These include the programs NROFF and TROFF for printing formatted documents on line printers and electronic type-setting equipment. There are also some useful "macros", such as EQN for printing complex mathematical equations, and TBL for presenting information in a tabular format.

Since the above mentioned facilities are highly specialized, these "standard" UNIX facilities for formatting documents, as well as some of the general purpose programming and debugging tools, are available as separate packages (see UNIX Q&A).

A description of all of these new facilities is beyond the scope of this preview. I will, however, make mention of a few of the simpler, but useful utilities I had the opportunity to use.

One of the first utilities used in the tutorials is called "wc", for "word count". It simply returns the number of lines, words, and characters contained in a specified file, or list of files. If you do a lot of writing, or like to enter those types of contests requiring 2500 words or less, this utility would come in handy.

Of more generalized usage is a utility known as "grep" (for General Regular Expression Patterns). What it does is search a file for a specific text pattern. It would enable you, for example, to find all of the lines of text in a text file, or list of text files, that contained the expression "is obstreperous and recalcitrant." Or suppose that among the hundred or so letters you have stored on the disk there is a specific one written to a Mr. Xinu which you wish to review, but you can't recall the name of the letter file. You could have "grep" locate the letter file by a command like:

```
grep -l 'Mr. Xinu' /usr/joe/letters/*.*
```

This would list only the names of the files that contained a reference to Mr. Xinu.

Another handy utility is "calendar," which acts as a reminder service. It requires that there be a file called "calendar" in the current directory. Into this file you put entries that contain a month-day date, e.g., Dec. 7, december 7, or 12/7, followed by a description of what is to take place. When the "calendar" utility is called it will search the file for any lines containing today's or tomorrow's date, and print those lines. With an argument present, it will search the calendar files present in all users login directories, and send any positive results to them via the "mail" utility. There is also a utility known as "cal", which when given an argument such as a month and year, will display a calendar for that period. If only a year is specified, it will generate a display showing the monthly calendars for that year.

The utilities described above represent a mere sampling of what is available. But the UniPlus + UNIX offers considerably more than simple utilities. For one, the shell is not only interactive, but it is fully programmable. Flow-of-control constructs such as IF-THEN-ELSE, WHILE, FOR, and CASE are present to facilitate sophisticated user programs consisting of UNIX commands or command scripts. In addition, the shell supports memory variables which can be used locally (in a given shell) or exported to other shells (a global variable). Thus it is possible to interactively program the shell to perform a variety of tasks based upon varying conditions. When you include redirection of I/O and pipes within these programs, you have the means for creating very powerful procedures. The ability to interactively program the shell is something that would be nice in CROMIX.

Not only can the shell be programmed, but the local operating environment can also be "programmed." The "mode" utility of CROMIX approaches this capability, but not to the extent of UNIX. For one, there is a TERMCAP file in which the various terminal characteristic definitions can be placed. So UNIX can accommodate a great variety of terminal types. Terminal definitions not contained in the stock TERMCAP file can be specified. Another useful feature is provided by the $PATH variable. This allows the user to dictate where UNIX should search for programs. In CROMIX, the search is rigidly defined: first the current directory is searched, and if the search fails, the /BIN directory is searched, followed if need be by searching the /CMD directory. UNIX works in a similar fashion except that the user can specify other directories to be searched. All of this environmental customization can be defined and automatically set-up during login by placing the appropriate commands in a file called .PROFILE. UNIX will execute this file, if present in the user's login directory, as soon as a user logs on. The method is comparable to CROMIX's execution of a /ETC/STARTUP.CMD file, except that a .PROFILE file can be made for each user.

The documentation alone probably weighs more than the computer system.

another nice feature, although one we did not get a chance to see demonstrated, is the inclusion of software to support an ETHERNET based local area network. This network utilizes the TCP/IP protocol used in ARPANET and Defense Data Network. To function, it is necessary that certain hardware (the ENET board pair and ETHERNET transceiver) and B-NET software be included in the system. Also included, and not requiring additional hardware (other than a modem) are a set of utilities for telecommunications called PUT and TAKE. These are not part of the standard UNIX, but were designed for UniPlus + operating in a 68000 environment, and allow for file transfers between UNIX machines. They are roughly equivalent to the RFILE and SFILE utilities of CROMIX.

DOCUMENTATION

Besides the online training made available to the participants of the seminar, we were all supplied with the complete set of documentation for UniPlus + System V. The majority of this is derived from the original Bell Labs UNIX documentation, with additional annotations on utilities specific to the UniPlus + V implementation. The documentation alone probably weighs more than the computer system. It is awesome, but considering that it originated from Bell Labs, to be expected. That is to say it is complete. On the other hand, except for some of the introductory articles, it is pretty high-level stuff (again to be expected from Bell Labs). It seems that each utility has a multitude of options and possible parameters, each of which are covered in seemingly excruciating detail—so much so that the obvious and intended use of the utility sometimes gets buried away.

There are some very nice features in the documentation, such as the "permuted index." This is a computer generated cross-reference index ("ptx" is the responsible utility) that greatly simplifies the process of locating where, within the massive set of literature, information on a specific topic can be found. The format is different from what you may be used to: three columns of information are presented. You search the middle column (which is alphabetically arranged) for a keyword relating to the topic in question. Once located, the pertinent section appears in the rightmost column. The leftmost column contains an extract of a sentence describing the topic which contained the keyword. Thus, many separate, but related, keywords will map to a single topic section. Another interesting feature is that there are no page numbers. All referencing is done by topic title, which are arranged alphabetically within the manuals.

The documentation we received consisted of an Administrators Manual and three additional volumes. The Administrators Manual deals primarily with the accounting utilities for system manage-

ment, and the various procedures involved in setting up a system, booting, and crash recovery.

Volume I is comprised of six sections. Section I is the heart of the Volume, and contains information on the general commands and applications: general purpose utilities, communication commands, and graphics commands. Section II deals with Systems Calls. Section III discusses Subroutines: C and Assembler Library Routines, Math Library Routines, and Standard I/O Routines. Section IV deals with the various File Formats, and Section V with Miscellaneous Routines. Everybody's favorite is Section VI, which is devoted entirely to the many games that come with the system (adventure, blackjack, backgammon, and trek [startrek], to name a few).

Volume II is equally as massive, with a section on Program Development Tools: the Shell Tutorials, UNIX Programming in 'C', YACC—Yet Another Compiler-Compiler, AWK—a Pattern Scanning Language, and LEX—a Lexical Analyzer Generator. Another section on Program Maintenance describes the use of the ADB Debugger and LINT, a 'C' program checker. The section on UNIX Maintenance and Information describes the FSCK utility for checking the UNIX file structure, the PWB/UNIX Accounting System, and an article entitled "The Portable C Compiler." There is also a section dedicated to the topic of networking.

Volume III is made up of a series of tutorials written by the designers of UNIX. In addition, the numerous document processing programs, such as NROFF, TROFF, EQN, and TBL are completely described. What is more, the vast majority of the documentation is present online, and can be accessed by way of the MAN utility. There is also an online HELP function similar to that of CROMIX.

I observed and experienced enough to realize that UNIX has a better chance at viability in the race for standardization than many of its more recently developed counterparts.

CONCLUSIONS

You can only learn so much over a period of a few days. Especially so when the subject at hand is so large and complex. But I observed and experienced enough to realize that UNIX has a better chance at viability in the race for standardization than many of its more recently developed counterparts. This is because it has had time to mature, and grow to meet the needs of those that employ it. The process of maturation will no doubt continue, and most likely at an accelerated rate.

As it reaches into the micro world, and the business market, there will be a great demand for friendly software. This will no doubt be quick to emerge. All the necessary developmental tools are already there, as is a large reservoir of software designed for the minis and mainframes.

The operating system itself will have to become friendlier, as CROMIX did with the introduction of products such as the Menu Generator. In the meantime, there will probably need to be professional system administrators to oversee the utilization of the computer systems in normal business environments. But the benefits that accrue, from software portability alone, will expand the vistas of users the world over.

Space Glizard: A 32K Structured Basic Game for the C-10

By Arie Grapa
Mexico City, Mexico

Editor's Note:

Every now and then we receive something that is not only useful, but fun. Such is the case with the program we received from Arie Grapa, of Mexico. It is written in 32K Structured BASIC for use on the C-10, and employs techniques for manipulating the graphic character sets of the C-10. We have run the program and it performs admirably. The listing herein provided was obtained directly from the source program, so if you enter it exactly as given it should work without error. We hope that other C-10 users will find it as entertaining and educational as we have. For those of you with an aversion to typing, and equipped with a modem, a copy of the listed version of the program is available from the CrosNest BBS (see OUTPUT). Arie Grapa developed this program on his own without the benefit of the C-10 Technical Reference Manual (which had yet to be published). He is eleven years old. Nice work Arie!

```
10   Rem GAME-SPACE GLIZARD
20   Rem AUTHOR-ARIE GRAPA - 11 YEARS-MICROMEX S.A. DE C.V.
30   Rem MEXICO - DECEMBER 1983
40   Set 0,-1
50   Lifes=5
60   On Esc Goto 1050
70   Integer Bonus,Aster'1,Aster'2,Aster'3,Aster'4,Aster'5
80   Esc$=Chr$(27)
90   Randomize
100  @ Esc$;"E"
110  @"WELCOME TO ''SPACE GLIZARD'',DO YOU WANT INSTRUCTIONS?"
120  Input Answer$
130  If Answer$="YES" Then Gosub 1310
140  @ Esc$;"Z"
150  Horizontal=15 : Vertical=15
160  Letter=0 : Time=Time+1 : If Time=100 Then Goto 1410
170  Num=Rnd(1)*10 : If Num(4 Then F=1
180  Rem HERE IT STARTS
200  Gosub 1470
210  Rem HERE IT MAKES THE FRAME
220  @ Esc$;"E";
230  @ Esc$;"R";
240  @ Esc$;"F";Chr$(31+3);Chr$(31+1);"ZbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbV"
250  @ Esc$;"F";Chr$(31+17);Chr$(31+1);"ZbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbV"
260  @ Esc$;"F";Chr$(31+1);Chr$(31+1);
270  @"BbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbF"
280  @ Esc$;"F";Chr$(31+16);Chr$(31+2);"                                                                          "
290  @ Esc$;"F";Chr$(31+17);Chr$(31+2);"cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc"
300    For Qq=2 To 23
310    @ Esc$;"F";Chr$(31+Qq);Chr$(31+1);"f"
320    @ Esc$;"F";Chr$(31+Qq);Chr$(31+76);"f"
330    Next Qq
340  @ Esc$;"F";Chr$(31+23);Chr$(31+1);"JbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbN";
350  @ Esc$;"S";
360  Rem THE FRAME HAS BEEN DONE
370  If Time=1 Then Aster'1=10 : Aster'2=20 : Aster'3=30 : Aster'4=40 : Aster'5=50
380    For Line=5 To 16
390    On Esc Goto 1180
400    @ Esc$;"R";
410    F=1
420    @ Esc$;"F";Chr$(31+17);Chr$(31+1);"ZbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbV"
430    @ Esc$;"S";
440    @ Esc$;"F";Chr$(31+2);Chr$(31+3);"  TIME:";Time;"     ";"POINTS:";Points;"          ";"LIFES:";Lifes
450    @ Esc$;"R";
460    @ Esc$;"F";Chr$(31+16);Chr$(31+2);"                                                                          "
470    @ Esc$;"F";Chr$(31+17);Chr$(31+2);"cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc"
480    @ Esc$;"S";
490    If Horizontal=16 Then Horizontal=19
500    If Horizontal=18 Then Horizontal=15 : @ Esc$;"F";Chr$(31+18);Chr$(31+Vertical);" "
510    If Horizontal=22 Then Horizontal=5 : @ Esc$;"F";Chr$(31+22);Chr$(31+Vertical);" ";
520    If Horizontal=4 Then Horizontal=21 : @ Esc$;"F";Chr$(31+4);Chr$(31+Vertical);" ";
530    If Vertical=2 Then Vertical=72 : @ Esc$;"F";Chr$(31+Horizontal);Chr$(31+2);" "
540    If Vertical=73 Then Vertical=3 : @ Esc$;"F";Chr$(31+Horizontal);Chr$(31+73);" "
550    Rem THE MAN MOVES TWICE AS FAST THAN THE ASTEROIDS
560    Rem WE REPEAT TWICE THE FOLLOWING
570    Z=Horizontal : X=Vertical
580    Rem THE FOLLOWING INSTRUCCTION GETS A SINGLE CHARACTER
590    Letter=Inp(52)
600    If Letter=77 Then Horizontal=Horizontal+1
610    If Letter=74 Then Vertical=Vertical-1
620    If Letter=75 Then Vertical=Vertical+1
630    If Letter=73 Then Horizontal=Horizontal-1
640    @ Esc$;"F";Chr$(31+16);Chr$(31+2);"
650    If Horizontal=16 Then Horizontal=19
660    If Horizontal=18 Then Horizontal=15 : @ Esc$;"F";Chr$(31+18);Chr$(31+Vertical);" "
670    If Horizontal=22 Then Horizontal=5 : @ Esc$;"F";Chr$(31+22);Chr$(31+Vertical);" ";
680    If Horizontal=4 Then Horizontal=21 : @ Esc$;"F";Chr$(31+4);Chr$(31+Vertical);" ";
690    If Vertical=2 Then Vertical=72 : @ Esc$;"F";Chr$(31+Horizontal);Chr$(31+2);" "
700    If Vertical=73 Then Vertical=3 : @ Esc$;"F";Chr$(31+Horizontal);Chr$(31+73);" "
710    @ Esc$;"F";Chr$(31+Horizontal);Chr$(31+Vertical);Esc$;"dH";Esc$;":";Chr$(6);Esc$;"p";Esc$;"d0"
720    @ Esc$;"F";Chr$(31+Z);Chr$(31+X);" ";
730    Z=Horizontal : X=Vertical
740    Letter=Inp(52)
750    If Letter=77 Then Horizontal=Horizontal+1
760    If Letter=74 Then Vertical=Vertical-1
770    If Letter=75 Then Vertical=Vertical+1
780    If Letter=73 Then Horizontal=Horizontal-1
790    @ Esc$;"F";Chr$(31+Z);Chr$(31+X);" ";
800    @ Esc$;"F";Chr$(31+Horizontal);Chr$(31+Vertical);Esc$;"dH";Esc$;":";Chr$(6);Esc$;"p";Esc$;"d0"
810    @ Esc$;"F";Chr$(31+Z);Chr$(31+X);" ";
820    Rem NOW WE MAKE THE ASTEROIDS
830    @ Esc$;"F";Chr$(31+Line);Chr$(31+Aster'1);Esc$;":";Chr$(2);Esc$;"p"; : If Line=Horizontal And Aster'1=Vertical Then 1070
840    If F=1 Then @ Esc$;"F";Chr$(31+Line);Chr$(31+Bonus);Esc$;":";Chr$(19);Esc$;"p";
850    If Line=Horizontal And Bonus=Vertical Then Dummy'2=1 : Gosub 1420
860    @ Esc$;"F";Chr$(31+Line);Chr$(31+Aster'2);Esc$;":";Chr$(2);Esc$;"p";
870    If Line=Horizontal And Aster'2=Vertical Then Goto 1070
880    @ Esc$;"F";Chr$(31+Line);Chr$(31+Aster'3);Esc$;":";Chr$(2);Esc$;"p"
890    If Line=Horizontal And Aster'3=Vertical Then Goto 1070
900    @ Esc$;"F";Chr$(31+Line);Chr$(31+Aster'4);Esc$;":";Chr$(2);Esc$;"p"
910    If Line=Horizontal And Aster'4=Vertical Then Goto 1070
920    @ Esc$;"F";Chr$(31+Line);Chr$(31+Aster'5);Esc$;":";Chr$(2);Esc$;"p";
930    If Line=Horizontal And Aster'5=Vertical Then Goto 1070
940    @ Esc$;"F";Chr$(31+Line-1);Chr$(31+Aster'1);" ";
950    @ Esc$;"F";Chr$(31+Line-1);Chr$(31+Aster'2);" ";
960    @ Esc$;"F";Chr$(31+Line-1);Chr$(31+Aster'3);" ";
970    @ Esc$;"F";Chr$(31+Line-1);Chr$(31+Aster'4);" ";
980    @ Esc$;"F";Chr$(31+Line-1);Chr$(31+Aster'5);" ";
990    @ Esc$;"F";Chr$(31+Line-1);Chr$(31+Bonus);" ";
1000   Next Line
1010 F=0
1020 On Esc Goto 1050
1030 Gosub 1470
1040 Goto 380
1050 @ Esc$;"Z" : Goto 1180
1060 Gosub 1470
1070 @ Esc$;"E"
1080 Gosub 1570
1090 Goto 1100
1100 @"BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB"
1110 Gosub 1570
1120 @"0000000000000000000000000000000000000000000000000000000000000000000"
1130 Gosub 1570
1140 @"MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM"
1150 Loses=Loses+1
1160 @ Esc$;"9"
1170 Lifes=Lifes-1 : If Lifes()0 Then Gosub 1470 : Goto 200
1180 @ Esc$;"d0" : @ Esc$;"S"; : @"DO YOU WANT TO PLAY AGAIN?" : @ Esc$;"Z";
1190 Input Answer$
1200 If Answer$="YES" Then @ Esc$;"Z"; : Gosub 1470
1210 If Answer$="YES" Then Goto 200
1220 If Answer$="NO" Then 1240
1230 Goto 1180
1240 @"AT LEAST DO YOU WANT TO KNOW HOW MANY TIMES YOU LOST OR WON?"
1250 Input Answer$
1260 If Answer$="NO" Then Goto 1300
1270 If Answer$="YES" Then @ Esc$;"Z"; : Goto 1290
1280 Goto 1240
1290 @ Points;" WON ";Loses;" LOST" : @ Esc$;"Z";
1300 End
1310 @"  In this game you have 5 lifes and you are a""";Esc$;"dH";Esc$;":";Chr$(6);Esc$;"p";Esc$;"d0";
1330 @""" and some Asteroids fall" : @"upon you as """;Esc$;":";Chr$(2);Esc$;"p";"""";
1340 @"but you have to try not to crash into them.For that, there"
1350 @"is a line in the frame and under that line no asteroid hits.To move"
1360 @"yourself, you use I(UP),M(DOWN),K(RIGHT),J(LEFT) and any other to stop."
1370 @"To get points you have to hit the""";Esc$;":";Chr$(19);Esc$;"p";"""";"."
1380 @"PRESS ";Esc$;"dH";"ESC";Esc$;"d0";" TO FINISH AND"
1390 Input"PRESS RETURN TO START",Answer$
1400 Return
1410 @"TIME OUT,"; : Goto 1180
1420 Hits=Hits+1 : Points=Hits*100
1430 @ Esc$;"F";Chr$(31+13);Chr$(31+13);"YOU HAVE ";Points;" POINTS" : Dummy=0
1440   For Dummy=Dummy To 2000 : Next Dummy
1450 @ Esc$;"F";Chr$(31+13);Chr$(31+13);"                              "
1460 Return
1470 Rem
1480 Rem HERE IT CALCULATES WHERE THE ASTEROIDS WILL FALL
1490 Bonus=Rnd(1)*69+4
1500 Aster'1=Rnd(1)*69+4
1510 Aster'2=Rnd(1)*69+4
1520 Aster'3=Rnd(1)*69+4
1530 Aster'4=Rnd(1)*69+4
1540 Aster'5=Rnd(1)*69+4
1560 Return
1570 @ Esc$;"8" :   For Dummy=1 To 500
1580   Next Dummy : Return
1590 On Esc Goto 1610
1600 Goto 110
1610 @ Esc$;"Z"; : Goto 1180
```

UNIX Q&A

Continued from page 11

management software, and many optional software utilities that are not necessary for the basic operation of the system. The swap space alone typically occupies four megabytes of disk.

15. How much disk can I add to a Cromemco system?

The STDC supports two five inch hard disks for a total of 100 megabytes and the alternate SMDI interface for SMD hard disks supports up to 1200 Megabytes of hard disks.

16. Now that we have UNIX V, what will CROMIX be used for?

CROMIX is a stable, proven product with a large Cromemco user-base. UNIX V is a widespread standard operating system that will attract a large base of portable software. From a user point of view the functions of the two operating systems may seem similar.

CROMIX was designed assuming:

1. RAM is inexpensive.
2. Disk Drives are slow.
3. Optimization for one to three users.
   a. Expansion through networking
   b. Workstation oriented
4. Flexibility needed for peripheral interfacing:
   a. High speed graphics
   b. Hard accelerators
5. Control by Cromemco desirable for customizations.

UNIX V, by comparison, seems to assume:

1. RAM is expensive (Swapping to save RAM usage)
2. Disk storage is inexpensive. (Requires larger disk than CROMIX)
3. Optimization for many users:
   a. Memory sharing for speed and RAM minimization
   b. Swapping for RAM minimization
4. Difficult interfacing and performance tuning
   a. Memory mapping complexity
   b. Size related complexity (CROMIX is smaller).
5. Control by AT&T to promote standardization.

17. How should I choose between CROMIX and UNIX?

In a sense, you don't have to choose between CROMIX and UNIX. When you purchase a Cromemco UNIX system you automatically get them both. The act of switching operating systems is as easy as issuing log-in instructions to the other operating system. Thus the two systems co-reside on the same hard disk and act as two partitions of the same system.

You still must elect to use one or the other for a specific purpose. This choice may be based on:

a. Which operating system the software to be run was written for:
   1. If it was written for CROMIX or CP/M you use the CROMIX partition.
   2. If it was written for UNIX you use the UNIX partition.

b. UNIX has protection features that are not in CROMIX:
   1. This may be desirable for use in high security applications and in schools where mischievous users may be a problem.
   2. The presence of this protection makes direct I/O from programs difficult. Therefore it is easier to write special purpose hardware-intensive code for CROMIX.

c. UNIX does swapping at times which are difficult to predict.
   1. The response time of UNIX varies with the operating system swapping.
   2. CROMIX does no swapping and therefore has a more predictable response time.

d. UNIX requires mapped memory.
   1. The exact hardware location in memory that a given program resides is difficult to control or determine. The philosophy is that the programmer doesn't need to know and if he does he shouldn't use UNIX.
   2. Memory mapping is done in software in CROMIX. Programs are run at the same logical address as their physical address. The program loader does the final linking as they are loaded off the disk. This makes program loading more controlled and predictable for special applications.

e. UNIX will not run without a hard disk.
   1. UNIX will not run from the floppy disk and cannot be booted from a floppy disk.
   2. CROMIX can be booted from a floppy. The RAM DISK feature in CROMIX allows extra RAM to be used as a high speed disk. Thus a CROMIX system can run with solid state reliability and disk reliability limitations. Booting from floppy is also useful for debugging the system.

f. The Maximizer (TM) 12 MIPS coprocessor currently only supports CROMIX.

g. The Cromemco D series computers are available at lower cost than the X-Series. These computers do not include the XMM hardware and can be used to provide excellent performance for those users that do not require UNIX System V.

(Editor's note: This Q & A session with Dr. Roger Melen will be continued next issue.)

Let CROMIX Do The Work Next Time

by Mark M. Byrd

There is an immediate question a user needs to answer whenever a new program is written for CROMIX. The question is: "Am I writing code that already exists in another CROMIX utility?"

As time is our most precious resource, if we can avoid rewriting code already provided by CROMIX, we solve two problems. First, the user is most likely familiar with the CROMIX utility, so it should be unnecessary to go into great detail on its use. And, second, less time will be needed to complete the final program. We recently had the opportunity to put this theory to work in one of the projects we were developing.

We have been supplying large capacity Winchester drives to our CROMIX customers for some time, and it was becoming painfully obvious that these customers were unsatisfied with using floppies as their only means of backup. A search was begun for an alternative to the floppies.

Today's technology being what it is, we concluded that tape drives might provide the only cost-effective solution. After selecting a manufacturer of 3M cartridge tape drives, the task at hand was to write the program to do the backup.

Several design goals were defined in order for the product to be acceptable to our customers. First, it had to be a "**hands off**" operation insofar as users were concerned. All customers said that their main backup complaint was the time element involved for the **operator.** They did not necessarily require that the backup be particularly fast, but they hated the idea of an operator having to **babysit** the system, rather than being productive at another task.

Second, the program should allow the user to backup and restore on a **file-by-file** basis. Most users, in our customers' experience, need to restore only one file, or group of files, due to accidental erasure or restoration of archived data.

Third, the user should be able to find out just what files are on a tape. Obviously, a **directory** of each tape became a critical part of the program.

Finally, users should be able to **customize** the program so that specific backup requirements could be carried out by unskilled personnel.

Our first task was to write the program so the user could specify a file, write that file to tape, and later be able to read the file back from tape to disk. At the same time, the user should be able to do a directory of a list of files on that tape. Writing this part of the program consumed the majority of our programming effort. Once this had been accomplished, the only requirement was to repeat the process for all the files the user wished to backup.

At that point, the question for us became: "Do we provide the code to specify which files to backup, or do we let CROMIX do the job?" This was the point at which we decided to provide a **shell command** to accomplish the task. CROMIX provides a very powerful utility called **FIND,** and this was just the utility needed. With this knowledge in hand, we began our first pass at the shell command.

Our logic was that FIND would build a file which contained a list of filenames that were to be backed up. The tape backup program would then read this input file and backup the specified files. Since the shell command knew which files were being backed up, the next logical step was for the shell command to create the reverse of backup. So, as the shell was creating the input file, the TAPE RESTORE command file was being created concurrently. This logic is what we ship today, but the results of our first attempts were less than satisfying.

Our mistake was that when the FIND command found a file that met the criteria, that same FIND command called another shell. The time required for this process on the Cromemco HD-20 was five hours. Even though this met the "hands off" requirement, it simply was not acceptable. After writing three lines in the shell command so that the FIND command did not call another shell, the same process took only ten minutes—certainly within acceptable limits.

The shell command appears to the tape backup unit as an interactive front end. It reminds the user to make certain the tape is inserted, provides the ability to get a hard copy of all files being backed up, builds the input file and, finally, performs the backup. By providing these capabilities in the shell rather than in the program itself, the user gains greater flexibility to tailor different versions of the shell to provide backup of specific directories.

Our first version would fill one tape and then stop. That was technically okay, except that our customers felt it would be better if the files could span tapes. So, with a small modification, the program now prompts the user for the next tape if the first tape becomes full.

Today, we supply this package to dealers and end-users, and everyone who uses it is glad it was developed. The time required for backup, including time to build the input file, is 5.5 minutes/megabyte. By using 555' tapes, up to 17.5 megabytes per tape can be achieved. It is only with the blend of custom programming, CROMIX utilities, and the shell command that this product is meeting with such success.

About the Author

Mark M. Byrd is a systems consultant and the owner of Microcomputer Consulting Services [MCS] in Richardson, Texas. Byrd has a degree in Electrical Engineering and spent the first five years of his career designing microcomputer hardware for the medical industry. He visited Cromemco's plant in 1977, and has been involved with their products since, becoming the first CROMIX user in Dallas. In 1981, Byrd formed his own company which strongly promotes Cromemco products in the Dallas/Ft. Worth area. He has developed numerous applications for CROMIX systems, specializing in "... they said it couldn't be done..." types of projects. Byrd can be reached by phone at (214) 699-7783.

WORKAHOLICS: KICK THE HABIT WITH SCADA.

How long do you think it would take you to analyze a 10-story building?...Two days?...Perhaps even four days? Now you can analyze it in just 15 minutes with our SCADA/Junior and C-10 combination, for a one time investment of $2995.00.

INTRODUCING SCADA/JUNIOR AND THE C-10.

SCADA/Junior is a minimized version of the original SCADA, our most popular and most powerful program, redesigned to work with the Cromemco C-10 Personal Computer.

The Cromemco C-10, as offered by American Computers & Engineers, comes with a Z-80 processor, 64K of RAM, and two 5¼" drives for a total capacity of 750K. The C-10 comes complete with a screen, a keyboard, and a four-program software library consisting of a word processor, financial spreadsheet, an investment planner and Cromemco's powerful 32K structured BASIC.

The combination is a complete and powerful engineering workstation that will dramatically increase your computational power and significantly decrease the time spent on routine and approximate calculations.

All for $2995.00.

FUTURE EXPANSION IS A BREEZE.

We know that once you start using SCADA/Junior, you will want to expand your capabilities further by using the full SCADA program which will allow you to handle: Thousands of degrees of freedom, response spectrum, time history, graphics, shear walls, shells, steel and concrete design.

American Computers & Engineers has a Cromemco/SCADA solution for all your growing needs such as a top of the line Cromemco System III which can handle 4 megabytes of RAM and a floating point processor working under a dual Z80/68000 8/16/32 bit processor for fast program execution. This system will accommodate hard disks and floppy drives; support up to eight users simultaneously; and run the full SCADA package to your greatest satisfaction.

WE MAKE IT EASY FOR YOU.

SCADA/Junior is our way of introducing you to the power of SCADA. Soon after you start using it, you will want to try the far greater capabilities of the full SCADA package. We make it easy for you by applying the full retail price of SCADA/Junior ($895.00) toward the purchase of SCADA/A.

ENGINEERS WORKING FOR ENGINEERS.

Take advantage of our expertise in computerized engineering applications. It won't cost you anything to find out how we may be of assistance. Just write to us or call our nearest office for details.

AMERICAN COMPUTERS & ENGINEERS

2001 So. Barrington Ave., Suite 204
Los Angeles, California 90025

LOS ANGELES:	(213) 477-6751
BERKELEY:	(415) 849-0177
SEATTLE:	(206) 583-0130
SPOKANE:	(509) 838-9800
PARIS, FRANCE:	285-46-40
GLASGOW, SCOTLAND:	248-2427
ATHENS, GREECE:	671-9722

THE CROMEMCO C-10 AND SCADA/JUNIOR $2,995.

In Case You Missed Something . . .

Back issues of I/O News are still available . . . $10.00 (U.S.)* per issue.

FEATURES & TOPICS

VOLUME I, NUMBER 1

16 FDC Controller Card
The CROMIX Operating System
The Cromemco Story
CDOS Active Command File
Intro to Computer Graphics
Summagraphics Bit Pad One: Configuring for Use With SDI
Bits & Bytes: System Conversion

VOLUME I, NUMBER 2

Cromemco System Zero
The C-BUS, IOP, and QUADART
Contract Issues for Purchasing a Computer System
Update on CDOS ACF
Structured Programming BASIC
Process Control in Petroleum Industry
Lawyers Service Billing
BASIC Disk File I/O
Announcement of RGB-13 Color Monitor
Intro for PS-8 Power Supply
New Diagnostics Available
Output: Your First Computer
Books — BASIC FORTRAN
Bits & Bytes: Shugart Drive Conversion

VOLUME I, NUMBER 3

Cromemco's C Compiler
BASIC String Arrays
Linear Static Analysis of Engineering Structures—Stress
FORTRAN Subroutines
Intro to the C Language
Software Update Service — SUDS
Estimating Taxes — Program
Error Re CDOS Disk Calls

VOLUME I, NUMBER 4

IOP/QUADART Phone Line Communications
Generate KSAM Files Using STD Data Base SCHEMA Bookkeeping
CDOS Command File Library
Computers in Psychology Lab
PASCAL for CROMIX
CROMIX and Communications Aid Construction Co.
Tec Tips: Old 4FDC Support DS Drives
Check for CRT's and Cables
16FDC Needs Jumper for 20MA OP Cables
Check Power Supply Voltage
Bits & Bytes: Heating Problem Solved
Estimate Taxes and Save (Program Correction)
Table of Cromemco Port Assigns
Graphics on 3355A
3102 Field Repair Notes
Double Density Conversion

VOLUME I, NUMBER 5

WRITEMASTER
Polynomial Interpolation
Using D.C. Hayes Micromodem
Message Utility Under CDOS
Mapping Routines for Floppy Disk Clusters
International Group Treasury System
Tec Tips: Hard Disk Warm Up
16FDC Cramped for Space
64KZ in Hot Systems
Using RDOS Diagnostics
Checking Ribbon Cables
Bits & Bytes: More on Retrofit Fans
CROMIX Utilities Offered

VOLUME I, NUMBER 6

Slidemaster & Fontmaster
Computers in Architecture
COBOL Program to Track Investments
Software Notes
New Enhancements to CROMIX
Encode/Decod for Basic String Functions
COBOL Programming
COBOL Programmers Aid (CPA)
Artwork Graphics and Text (COMANDER I)
Tec Tips: 3102 Has Loose ROMS
3355A Has Self-Test
Z-2H Power Plugs
CROMIX With Hard Disks
Reversing Fans
Use Only 3 Wires on 4FDC
Z-4D?
Bits & Bytes: CP/M Screen Editor Found
Trip From Cromemco
SUPERCOPY II

VOLUME II, NUMBER 1

System One Intro at WESCON
Structural Analysis Program (SAP-80)
CP/M Simulator
DP Results on Site at a Footrace
Spool Routine for 32K SBASIC
Extending Your Control Over Applications Software
Cromemco Overlay Linker
RBTE: Software for Binary Synchronous Protocol
Tec Tips: Shorted Tantalyn Capacitors
Keyboard Connect to TUART
PerSci 299 Head Damage
Early Rev. 64KZ's & CROMIX
Z-2H Power Plug Problem
Stop Auto Form Feed in CROMIX
Fast vs. Slow Seeks
Priority Interrupt Chaining
Location of 64KZ to WDI
Bits & Bytes: 32K SBASIC File Error: CROMIX
SUDS Review
Special Characters in COBOL
The COBOL Corner: COBOL: Using Segmentation

VOLUME II, NUMBER 2

Cromemco's New European Subsidiary
Overlaying Program & Data in Cromemco 16K BASIC
Emulating CP/M: Under CDOS & CROMIX
System III in Medical Lab
SpellMaster
Planning & Control on Micro
CROMIX Using 16K or 32K BASIC
Input: Shugart vs. Seimans Drives
Output: Group Insurance a Reality
Hybrid Equipment
Tec Tips: Hard Disk Notes
Fast Copy Programs & XFER
Office Static Control
Loose Chips
Stat- Label Features
Fast Memory Chips
Ground Fault Problems
3355 Spinwriter Driver
PRI Revision B
Oscillators on TUARTS
Cleaning Edge Connectors
Bits & Bytes: CROHORTS Building Library
SUDS for COBOL Users
CPMSIM Update
Printer Conflict Resolved
The COBOL Corner: Programming in COBOL

VOLUME II, NUMBER 3

Hard Disk for Systems 2 & 3
Local Area Network for Micros
Supercharge FORTRAN: APU
Implementation of Clock/Calendar Board
Review of dBASE II
Programming in C
Enhancements to RATFOR
Input: CROMIX with CDC 96 Meg Disk
Output: PL/1 Used Under CDOS
Insurance in Force
Tec Tips: Bringing Up Other Terminals
Serial Printers in CROMIX
Termination Resistor Packs
Using Special Printer Features
Bits & Bytes: Statistical Software
CP/M Re-Execution Under CDOS
IDS 88-Modem Modification
SUDS Updates Due
The COBOL Corner: Observations on COBOL 4.60

VOLUME II, NUMBER 4

Cromemco Expands Product Line: 68000 Based Boards
Putting Data Files in Order: Using DBMS
CDOS-CP/M Simulator (GEI)
SCADA: Structural Design Analysis
Parallel Port Adapter (PPA)
How the DPU Works
A Review of Planease
Automated Materials Handling
Input: Bringing Up Other Terminals
Connect REMEX or QUME Drive
Output: Casualty Insurance in Force
Tec Tips: Increase Time on 16FDC Timer
Printer Test Using BASIC
Printer Problems in CROMIX
Special Software Needed?
Bits & Bytes: Statistics Software Emerging
4PIO Boards Revisited
Australian Educational Group

VOLUME II, NUMBER 5

C-10 Unveiled at NCC
Interrupt Processing Tutorial
COGO: Civil Engineering Program
System in Occupational Safety
Micro Based Radar System
Budget Graphics for Z-2D
Input: CROMIX 11.05 & TUART Rev. E
Statistics Software
IFDAS: Data Analysis System
Output: CDOS & CROMIX Drivers
FORTH
Soft Tips: User Memory Under CDOS
Handy Command Files
CROMIX Mode Table
Bits & Bytes: Disk Insertion in 299B Drives
New Games Package
Interface for Diablo Printer
32K Classroom: Absolute Address Cursor Control

VOLUME II, NUMBER 6

256K Card Lowers Memory Cost
Overview of CAD/CAM
An Associative Data Base
Breakthrough in Robotics

*Order from I/O News by Volume & Issue Number. Price includes air mail postage.

Cromemco Intros 8 New Boards: CTI SDD C-Net
DAC-12 ADC-12
Terminal Attributes Thru "C"
For/Next Statements Using Variable Steps
Recovering Erased Disk Files
High Level COBOL Compiler
Variable Expressions for Calculating SBASIC Strings
Input: Initializing Disks Under cdos
Comet-FORTH for CDOS & CROMIX
Removing Tarnish from IC Pins
Output: WESCON & MINI/MICRO '82
Tec Tips: Bringing Up CRTs in CROMIX
Diagnostic Package
RPM Adjustments for Disk Drive
Power Plug in Hard Disk Drives
Using Tape to Test Cards
Ground Straps on Hard Disk
New 64KZ-II
Bits & Bytes: X-ON, X-OFF Protocol Patch
SS Drives & DS Diskettes
32K
Classroom: CRT Terminal Graphic Mode

VOLUME III, NUMBER 1

System Three Gets New Look
Tips on Using Your C-10
An Associative Data Base (Part 2)
Micro Operation w/o Down Time
Conditional for CDOS .CMD File
A New Approach to Statistical Econometric System
Directory Check for FORTRAN
Review of IBM2CPM.COM
New 32K BASIC Book
Input: Printer Form Feed—In BASIC
System 3 Key Problem
Correction to Input
CROMIX Drivers Coming
Output: Watch Your Power
Tec Tips: Serial Printers/TUART/CROMIX
Modems with CROMIX
CPU Selection on WDI-II
PRI Modifications for CROMIX
New C-10: Brief Review
Dead PerSci 299 Drives
Hard Disk Initialization
Soft Tips: Repeat/While Loop in 16K
Organizing Files in CROMIX Dir
32K
Classroom: Single Letter Selection Program
Printing A CRT Display: BASIC

VOLUME III, NUMBER 2

20 Megabyte Disk Now Available
Direct I/O in "C" Using In-Line Assembly Code
32K SBASIC Benchmark Test
Enhancements to the XFER Utility
C-10 Developments Discussed
Complex Arithmetic Library for FORTRAN
Improved Conditional for CDOS
Some Useful Patches for dBASE and Screen
New Cromemco Products
New Cromemco LISP for AI
GETEDIT: Terminal I/O Routine for CDOS & CP/M
Cromemco Intros Statmaster Statistics Program
Input: ^S to Halt BASIC Program
Hard Disk Warm-Up Before Boot
Output: Value of Education: Seminars
Tec Tips: HDD-5 Index Sensors
New 5 Inch Hard Disk: 20 Meg
CP/M and CROMIX Compatibility
Soft Tips: Provided Assembler Routines
Margins/Tabs for Screen.COM
Alternative to STARTUP.CMD
Bits & Bytes: EDIT Discoveries
Graphic Library for Cromemco
Help: CROMIX Process Table
Recovering Erased Files
32K
Classroom: Creating a File

VOLUME III, NUMBER 3

Image Processing of Satellite Data with Micros
Comparison of CROMIX and UNIX
Standby UPS Power Systems
More C-10 Developments
Jump Under RDOS or CDOS
X-ON/X-OFF Protocol Patch
Customizing The C-10 Menu
MICROCAD/ME: CAD/CAM for Microcomputers
Interfacing FORTRAN with CDOS
Input: Questions About CROMIX-D
Tec Tips: Serial Printer: CROMIX 11.16
Card Puller
Floppy Disk Controllers
Upgrade PerSci Drives: CS-3
Bits & Bytes: Hotline for QUME Printer Users
Set Internal Clock on C-10
Fontmaster & Other Graphics
New Math Software Offered
Correction to Program
Peculiarity in 16K BASIC
Problem with Foreign Terminal
Inside
CROMIX: Using Version 11.16 SIM.BIN
CMD File: Freeing Disk Space

VOLUME III, NUMBER 4

Micropro Software/IBM Keyboard for C-10
DPU Price Reduced
Growing Pains Eased by Dealer Support: Example
Simultaneous Queued Output to 2 Printers: CROMIX
Input: GRAPHTXT for Fontmaster
C-10 Graphic Character Sets
Output: CROMIX Seminars/dBASE Seminar
Tec Tips: Adding 2nd Floppy to C-10
2.56 CDOS Hard Disk Bug
Bug in CROMIX 11.16 Typ Driver
What is 3355 Printer?
Dial Up Bulletin Boards
Soft Tips: Group Utility
System Programmers Corner
CROMIX 11.16 Update
Handy Command Files: Using 'IF'
Bits & Bytes: Simula for D-Series Cromemco
New Book for WordStar Users
Architectural Software
COBOL Programmers Aide II
Petroleum Library: 68000 Sys
CP/M & SIG/M Libraries: Rent
Graphics Package: *Plot
Equine Management System
Stack Initialization in Assem
C-10 Technical Manual ($35)
Inside
CROMIX: Running CP/M CDOS Under CROMIX
32K
Classroom: String and File Handling

VOLUME III, NUMBER 5

SDD: TV Camera Interface for Cromemco Systems
CROMIX— A User's Guide
Cromemco 1984: An Interview with Dr. Garland
Using C-10 Computer Ports
Using dBASE II: Setting the System Clock
Software Review: The Menu Generator
Input: Intro to 32K SBASIC for C-10
PL/1
BSTAM Communications Program
Super Calc 2 Under CDOS 2.56
Tec Tips: Multiple Printers in CROMIX
Soft Tips: Control of Printer Modes
CROMIX Sys Calls to Ren File
Bug in HDTEST with DPU
CDOS Drives on CROMIX Systems
Patch to COBOL Under CROMIX
CROMIX Input & TESTINP Utilities
Bits & Bytes: Screen Patches for H-19
New Products from Cromemco
Inside
CROMIX: RAMDISK
32K
Classroom: Reading CDOS Disk Label

VOLUME III, NUMBER 6

Formatted Input for 32K SBASIC
Software Review: proBOOKS C-10 Accounting System
Close Encounters of the C-10 Kind:
New Column Devoted to C-10
Cromemco Network at Chicago Mercantile Exchange
What is SUDS & Who Needs It?
New Software for the C-10
Input: Response to Need for Statistics Software
Problems with CP/M Simulation
More Suggestions: Book Reviews
Spooling to Two Printers
Output: I/O News Mailing Labels: Renew
Modem in Use at I/O News
CROMIX — A User's Guide
Seminars & Cramps
Insurance: IACU EDP Policy
Tec Tips: New RDOS, Boards, Disk Drives
Soft Tips: passwd and startup.cmd
Applications Using Spool
Operating System Updates
Bits & Bytes: New Canadian Cromemco Dealers
Cramper Plans Crystallizing
Inside
CROMIX: testinp: Fact and Fiction
Management of CROMIX Systems
32K
Classroom: Procedures to Control Cursor
Use of KSAM in 32K SBASIC

We're Interested!

We're Really Interested...in what you have to say. Especially about how you use your system...the problems encountered and the solutions effected...unusual uses or environments...and any practical applications you would be willing to share with fellow members. These can be short notes for departments like 'bits & bytes...' and 'Tec Tips,' or full feature articles.

Contact Richard Kaye or Lynn Platzek at I/O News for editorial guidelines or assistance. We're interested in unleashing your literary talents.

bits & bytes, nibbles & tweaks

IACU Contributing Author Elected to Select Academy

Jerome J. Tiemann, Ph.D., a contributor of articles to I/O News since 1980, was recently elected to the prestigious National Academy of Engineering. Election to this body is the highest professional distinction that can be conferred on an engineer. It honors important contributions to engineering theory and practice, significant contributions to the literature of engineering, or unusual accomplishments in new and developing fields of technology.

Dr. Tiemann, who has more than 80 patents to his credit, is currently consulting professor and fellow at the Stanford Center for Integrated Systems while on sabbatical from the General Electric Research and Development Center in Schenectedy, New York.

Third Party Forth Available for 68000 CROMIX

There now exists a version of Forth for 68000 CROMIX as a result of a joint effort between IACU Member Stewart Wilson of Concord, Massachusetts, and a producer of Forth language systems. The new version permits full 32-bit addressing so that the vast memory capacity of Cromemco DPU systems can be exploited. Also, this Forth compiles both "screens" and free format ASCII source files. Interested readers should write Wilson at: 30 Lang Street, Concord, MA 01742.

Demand for Environmental Software

Dr. Enrique Grapa, Director General of Micromex, a major Cromemco distributor in Mexico, has need for software dealing with environmental controls. If you know of any software that functions well in this broad area of interst, please contact Dr. Grapa at Micromex, No. 10 -80 Piso, Col. Napoles, C.P. 03810, Mexico, D.F.

Utility Provided by CUG — Holland

A disk management system for CROMIX systems which was compiled by Ben Verloop, a member of the Cromemco Users' Group — Holland, is now available to IACU members. The set of utility programs can maintain an archive of all filenames, their sizes, and their location (on which numbered diskette they reside). It provides an online "directory" of all of your various files, be they on CROMIX or CDOS diskettes, which you can add to and query.

Because of the general applicability of this utility to a wide range of users, we offer it in its entirety following. (Note: The "A" with the two dots over it can be interpreted as a left bracket, and the "U" with the two dots over it cane interpreted as a right bracket.)

```
DISK MANAGEMENT SYSTEM
----------------------

This  system  allows  the user to maintain a file of all the names of
all the files on his/her disks in a cromix environment.  (small  and
large  Cromix  and  CDOS disks) For each file the following items are
recorded:

1. size of the file (ordinary file) / device number (device file) /
   number of entries (directory file)
2. Filetype: ' '-ordinary, 'D'-directory, 'B'-block device,
   'C'-character device
3. Number of links
4. Date/time last updated
5. Complete pathname, including diskname (DSNNN)

Using  this system, the user can easily allocate a file on one of the
disks.  This system contains several command files, and  two  patched
cromix utilities.  The information is kept in four files:

/archive/xs (cromix small disks)
/archive/xl (cromix large disks)
/archive/cs (cdos small disks)
/archive/cl (cdos large disks)

(In four separated files to keep the search time as low as possible)

Each of the disks gets a name: DSNNN
        O=Operating system: x=cromix, c=cdos
        S=Size: s=small, l=large
        NNN=Number of the disk

If  you  want  to add a new disk to the list, or you have worked on a
disk, and want to update the information file, type:

# diskadd (device) (disktype) (disksize) (disknumber)
        (device):     any cromix device name (sfda, ..., fdd)
        (disktype):   operating system (x=cromix, c=cdos)
        (disksize):   size of the disk (s=small, l=large)
        (disknuaber): number of the disk (NNN)

If you want to know  on  which  disk  the  file  named  'find.me'  is
located, type:

# disk ÄsizeÜ find.me
        ÄsizeÜ is optional:
        Ä-xÜ = cromix disks     Ä-xsÜ = small cromix disks
        Ä-cÜ = cdos   disks     Ä-xlÜ = large cromix disks
        Ä-lÜ = large  disks     Ä-csÜ = small cdos   disks
        Ä-sÜ = small  disks     Ä-clÜ = large cdos   disks
        I ÄsizeÜ is omitted, disk will check all four information
        files: xs, xl, cs, cl.

Installation
------------

Before you can use the system, it must be installed:

NOTES:  (ret)  means 'press return key'.  (ret) is at the end of each
        line for readability. press return after the last significant
        character of the line.
        (esc) means 'press escape key'.
        the text between stars is comment: *comment*

1. Make a copy of 'L.BIN' (version 00.11) and patch it:
# copy /bin/l.bin /bin/ll.bin                                   (ret)
# patch /bin/ll.bin                                             (ret)
)s71A                                                           (ret)
00071A: CF  00          *remove printing of the owners*         (ret)
00071B: 1B  00          *name (eg. system, bin, ...)  *         (ret)
00071C: E1  .                                                   (ret)
)s771                                                           (ret)
000771: CF  00          *remove printing of the access *        (ret)
000772: 1B  00          *mode (eg. rewa re-- re--, ...)*        (ret)
000773: 21  .                                                   (ret)
)e                                                              (ret)
#
2. Make a copy of 'MATCH.BIN' (version 00.03) and patch it:
# copy /bin/match.bin /bin/mmatch.bin                           (ret)
# patch /bin/mmatch.bin                                         (ret)
)sle3                                                           (ret)
0001E3:  CF  00         *remove printing of the filename*       (ret)
0001E4:  1B  00                                                 (ret)
0001E5   E1  .                                                  (ret)
)e                                                              (ret)
#
3. Make a directory for the information files: '/archive'
   and create the four information files.
# makdir /archive                                               (ret)
# create /archive/xl    *cromix large*                          (ret)
# create /archive/xs    *cromix small*                          (ret)
# create /archive/cl    *cdos   large*                          (ret)
# create /archive/cs    *cdos   small*                          (ret)
#
 4. Make two redirected input files for 'screen'. These are used
    to add the diskname to the list of names, that is made by
    cdoscopy.
# create /archive/screen.inout1                                 (ret)
# patch /archive/screen.inout1                                  (ret)
)s                                                              (ret)
000000:  00  69,20,0D,1B,68,23,73,2F,0D,2F,2F,0D                (ret)
00000C:  00  .                                                  (ret)
)e                                                              (ret)
# create /archive/screen.inout2                                 (ret)
# patch /archive/screen.inout2                                  (ret)
)s                                                              (ret)
000000:  00  3A,2F,0B,0D,64,0D,1B,6A,62,64,0D,1B,65,75          (ret)
00000E:  00  .                                                  (ret)
)e                                                              (ret)
#
 5. Make a command file: 'DISKADD.CMD'. This command file is used
    to make a OSNNN file of the names of all the files on a disk.
# screen /cmd/diskadd.cmd                                       (ret)
I       *insert mode*                                   * no (ret) *
%start                                                          (ret)
 if x = x#1 goto syntax                                         (ret)
 if x = x#2 goto syntax                                         (ret)
 if x = x#3 goto syntax                                         (ret)
 if x = x#4 goto syntax                                         (ret)
 if #1 = sfda goto driveok                                      (ret)
 if #1 = sfdb goto driveok                                      (ret)
 if #1 = sfdc goto driveok                                      (ret)
 if #1 = sfdd goto driveok                                      (ret)
 if #1 = fda goto driveok                                       (ret)
 if #1 = fdb goto driveok                                       (ret)
 if #1 = fdc goto driveok                                       (ret)
 if #1 = fdd goto driveok                                       (ret)
 goto syntax                                                    (ret)
%driveok                                                        (ret)
 if #3 = L goto sizeok                                          (ret)
 if #3 = S goto sizeok                                          (ret)
 goto syntax                                                    (ret)
%sizeok                                                         (ret)
 if #2 = X goto cromix                                          (ret)
 if #2 = C goto cdos                                            (ret)
 goto syntax                                                    (ret)
%cromix                                                         (ret)
 cre /#2#3#4)*/dev/null                                         (ret)
 mount #1 /#2#3#4                                              (ret)
 find /#2#3#4 -name '**' -o -name '.**' -a -exec newfile äü #2#3#4(ret)
 unmount #2#3#4                                                 (ret)
 del /#2#3#4)*/dev/null                                         (ret)
 goto done                                                      (ret)
%cdos                                                           (ret)
 cdoscopy -l #1 ) /archive/#2#3#4                               (ret)
 ty /archive/screen.inout1)/archive/screen.input                (ret)
 echo -n #2#3#4))/archive/screen.input                          (ret)
 ty /archive/screen.inout2))/archive/screen.input               (ret)
 screen /archive/#2#3#4 ( /archive/screen.input)*/dev/null      (ret)
 del /archive/#2#3#4.bak /archive/screen.input)*/dev/null       (ret)
%done                                                           (ret)
 echo 'Ready with disk in device #1'                            (ret)
 mmatch -r #2#3#4 /archive/#2#3)/archive/#2#3.temp              (ret)
 del /archive/#2#3.bak                                          (ret)
 ren /archive/#2#3 /archive/#2#3.bak                            (ret)
 ren /archive/#2#3.temp /archive/#2#3                           (ret)
 ty /archive/#2#3#4))/archive/#2#3                              (ret)
 del /archive/#2#3#4
 exit                                                           (ret)
%syntax                                                         (ret)
 echo 'Syntax error; please type:'                              (ret)
 echo 'diskadd ÄdeviceÜ ÄdisktypeÜ ÄdisksizeÜ ÄdisknumberÜ'     (ret)
 echo 'ÄdeviceÜ  :  any of sfda, sfdb, ..., fda, fdb, ...'      (ret)
 echo 'ÄdisktypeÜ:  X for cromix, C for CDOS/CPM'               (ret)
 echo 'ÄdisksizeÜ:  Size of the disk, S=Small, L=Large'         (ret)
 echo 'ÄdisknumberÜ: Number of the disk, tree digits'           (ret)
 exit                                                           (ret)
ÄescÜ           *stop the insert mode*
EU              *exit and update*
#
 6. Make a command file 'DISK.CMD'. This command file is used
    to find a name in one of the four information files.
# screen /cmd/diskadd.cmd                                       (ret)
I       *insert mode*                                   * no (ret) *
%start                                                          (ret)
 if x = x#1 goto syntax                                         (ret)
```

```
 if #1 = -s goto small                                          (ret)
 if #1 = -l goto large                                          (ret)
 if #1 = -x goto cromix                                         (ret)
 if #1 = -c goto cdos                                           (ret)
 if #1 = -a goto all                                            (ret)
 if #1 = -xl goto cromixlarge                                   (ret)
 if #1 = -xs goto cromixsmall                                   (ret)
 if #1 = -cl goto cdoslarge                                     (ret)
 if #1 = -cs goto cdossmall                                     (ret)
 match '#1' /archive/xl                                         (ret)
 match '#1' /archive/xs                                         (ret)
 match '#1' /archive/cl                                         (ret)
 match '#1' /archive/cs                                         (ret)
 exit                                                           (ret)
%small                                                          (ret)
 match '#2' /archive/xs                                         (ret)
 match '#2' /archive/cs                                         (ret)
 exit                                                           (ret)
%large                                                          (ret)
 match '#2' /archive/xl                                         (ret)
 match '#2' /archive/cl                                         (ret)
 exit                                                           (ret)
%cromix                                                         (ret)
 match '#2' /archive/xs                                         (ret)
 match '#2' /archive/xl                                         (ret)
 exit                                                           (ret)
%cdos                                                           (ret)
 match '#2' /archive/cs                                         (ret)
 match '#2' /archive/cl                                         (ret)
 exit                                                           (ret)
%cromixsmall                                                    (ret)
 match '#2' /archive/xs                                         (ret)
 exit                                                           (ret)
%cromixlarge                                                    (ret)
 match '#2' /archive/xl                                         (ret)
 exit                                                           (ret)
%cdossmall                                                      (ret)
 match '#2' /archive/cs                                         (ret)
 exit                                                           (ret)
%cdoslarge                                                      (ret)
 match '#2' /archive/cl                                         (ret)
 exit                                                           (ret)
%all                                                            (ret)
 echo /archive/xs                                               (ret)
 ty /archive/xs                                                 (ret)
 echo /archive/xl                                               (ret)
 ty /archive/xl                                                 (ret)
 echo /archive/cs                                               (ret)
 ty /archive/cs                                                 (ret)
 echo /archive/cl                                               (ret)
 ty /archive/cl                                                 (ret)
 exit                                                           (ret)
%syntax                                                         (ret)
 echo 'Please type: disk Äsizeü Äname to searchü'               (ret)
 echo 'Äsizeü is optional:'                                     (ret)
 echo 'Ä-xü = cromix disks      Ä-xsü = small cromix disks'     (ret)
 echo 'Ä-cü = cdos   disks      Ä-xlü = large cromix disks'     (ret)
 echo 'Ä-lü = large  disks      Ä-csü = small cdos   disks'     (ret)
 echo 'Ä-sü = small  disks      Ä-clü = large cdos   disks'     (ret)
 echo 'If you know the Äsizeü, use it.....'                     (ret)
 echo 'You can get a list of all the names by entering:'        (ret)
 echo 'disk -a'                                                 (ret)
%exit                                                           (ret)
Äescü           *stop the insert mode*
EU              *exit and update*
#
```

7. Make a command file 'NEWFILE.CMD'. This command file is used to append the information of a file to the OSNNN file.

```
# screen /cmd/newfile.cmd                                       (ret)
I       *insert mode*                                   * no (ret) *
%start                                                          (ret)
 if x = x#1 goto syntax                                         (ret)
 if x = x#2 goto syntax                                         (ret)
 ll -ld #1)>/archive/#2                                         (ret)
 exit                                                           (ret)
%syntax                                                         (ret)
 echo 'Something is wrong in the command file /cmd/diskadd.cmd' (ret)
%exit                                                           (ret)
Äescü           *stop the insert mode*
EU              *exit and update*
#
```

```
*** END OF INSTALLATION ***

USING THE SYSTEM
----------------

You  are  now ready to use the disk management system: Insert a large
cromix disk in drive fdb.  Type:
# diskadd fdb x 1 001                                           (ret)
Now search for the file (filename):
# disk (filename)                                               (ret)
....
....
....
#
You may now look in the directory /archive:
# l /archive                                                    (ret)
      12    1 screen.inout1
      14    1 screen.inout2
 209,958    1 xl
The first two are the screen redirected inout files. The last one
is the information file for all your large cromix disks. (in this
one, the information of 23 large disks is stored)

*** GOOD LUCK WITH YOUR DISK MANAGEMENT SYSTEM ***
```

Current Software Product Version Report

This table lists the current versions of all Cromemco software. It was derived from Cromemco's Software Product Version Report of July 15, 1984. The following notations are used: "NA" implies that the information is not applicable or was not supplied in the product version report. An "*" after the model or release number indicates a preliminary release. Models which have a "–D" indicate 68000 software. Almost all software is supplied on both 8 inch and 5 inch diskettes, so the "L" (for large) and "S" (for small) have been omitted from the model numbers. Also, almost all software is supplied on Double Sided, Double Density diskettes.

MODEL	PACKAGE	RELEASE	VERSION	DATE MASTER CREATED
ANI-D	ANIMATOR (COBOL-D DEBUGGER)	1	NA	10/24/83
AP	ACCOUNTS PAYABLE		02.65	01/11/82
AR	ACCOUNTS RECEIVABLE		02.65	01/11/82
ASM-D	68000 MACRO ASSEMBLER (CROMIX)	2	01.14	02/16/83
BAS-D	68000 BASIC	1	02.10	11/07/83
C10CPM	C-10 CP/M OPERATING SYSTEM	1	02.00	01/17/84
CAMR	CALCMASTER	4	NA	02/29/84
CCC	CROMEMCO 'C' COMPILER	2	05.10	01/04/83
CCC-D	68000 'C' COMPILER	3	02.15	12/08/83
CDS	CROMEMCO DIAGNOSTIC SOFTWARE	4	NA	07/15/83
CISAM-D*	C-ISAM	1*	1.02	06/13/84
COB-D	68000 COBOL COMPILER	1	NA	11/04/83
COLL	CROMEMCO OVERLAY LINKER	3	02.04	03/25/83
CRO-D	68000 CROMIX OPERATING SYSTEM	7	20.63	05/16/84
CROMIX	Z-80 CROMIX OPERATING SYSTEM	10	11.26	05/16/84
CROMIX*	Z-80 CROMIX PRELIMINARY RELEASE	11*	11.27	07/03/84
CSPD	C-10 SUPER PACK	5	NA	06/05/84
CXDR	CROMIX DRIVER PACKAGE	1	NA	05/18/83
DAZZLER	DAZZLER GRAPHICS SOFTWARE	NA	NA	07/08/80
DBM	DATABASE MANAGER/REPORTER	NA	03.05	01/08/81
DGR	DAZZLER GRAPHICS PACKAGE	NA	NA	07/07/80
DIMR	DISKMASTER	1	01.05	12/28/83
DIMR*	DISKMASTER	2*	01.07	07/05/84
DOS	CDOS OPERATING SYSTEM	12	02.58	11/07/83
FDA	Z-80 MACRO RELOCATING ASSEMBLER	12	03.10	07/18/83
FDB	Z-80 BASIC	11	05.70	03/29/83
FDC	Z-80 COBOL COMPILER	6	04.64	03/29/83
FDF	Z-80 FORTRAN COMPILER	11	03.42	03/30/83
FDG	GAME PROGRAMS	NA	NA	09/04/80
FDR	Z-80 FORTRAN WITH RATFOR	4	01.05	03/29/83
FM2-D	FORMS-2 (COBOL-D FORM GENERATOR)	1	NA	10/24/83
FOMR	FONTMASTER	5	01.16	08/19/83
FOR-D	68000 FORTRAN COMPILER	6	02.15	05/17/83
FSTBAS-D	68000 FAST BASIC	1	02.10	03/23/84
FSTCCC-D	68000 FAST 'C' COMPILER	1	02.10	12/15/83
FSTCCC-D*	68000 PRELIMINARY RELEASE	2*	02.15	07/05/84
FSTFOR-D	68000 FAST FORTRAN COMPILER	1	02.11	12/15/83
FSTFOR-D*	68000 PRELIMINARY RELEASE	2	02.15	07/05/84
FSTPAS-D	68000 FAST PASCAL COMPILER	1	02.10	12/15/83
FSTPAS-D*	68000 PRELIMINARY RELEASE	2	02.15	07/05/84
GL	GENERAL LEDGER PACKAGE	NA	02.61	01/11/82
IDS	IOP DEVELOPMENT SOFTWARE	6	03.00	07/25/83
IN	INVENTORY PACKAGE	NA	02.65	01/11/82
INFX-D	INFORMIX (68000 RELATIONAL DBMS)	1	03.11	04/24/84
KSAM	KSAM FILE ACCESS SYSTEM (CROMIX)	3	01.04	03/01/83
LSP	LISP	5	01.08	03/31/83
NET	C-NET NETWORK SOFTWARE	2	NA	03/20/84
PAS-D	68000 PASCAL COMPILER	5	02.15	05/17/84
RBTE	REMOTE BATCH TERMINAL EMULATOR	5	01.08	11/17/83
RPG	Z-80 RPG II COMPILER	4	03.02	03/31/83
SDDDEMO	SDD DEMONSTRATION SOFTWARE	1	NA	01/26/84
SDIDEMO	SDI DEMONSTRATION SOFTWARE	5	NA	04/04/83
SGS	SDI GRAPHICS SOFTWARE	6	02.00	09/06/83
SLMR	SLIDEMASTER GRAPHICS EDITOR	4	02.03	04/26/83
SMDS	SMD DRIVE CONTROLLER SOFTWARE	1	NA	02/27/84
SMDS*	SMD PRELIMINARY RELEASE	2	01.08	07/06/84
SPICE-D*	SPICE CIRCUIT DESIGN SOFTWARE	1*	11.02	06/21/84
SPMR	SPELLMASTER SPELL PROOFING PROGRAM	5	01.20	06/27/83
STB	32K STRUCTURED BASIC	11	03.65	03/31/83
STB-D*	68000 STRUCTURED BASIC	1*	3.65.04D	04/13/84
STMR	STATMASTER STATISTICAL PROGRAMS	2	01.04	07/21/83
TDS	TAPE DRIVE SOFTWARE	4	11.11	04/01/83
TEMR	TELEMASTER COMMUNICATIONS SOFTWARE	2	02.06	04/24/84
TSDI	TRI-SDI GRAPHICS SOFTWARE	3	NA	04/01/83
TSS	TRACE SIMULATOR	NA	02.06	03/06/80
WPS	WORD PROCESSING SYSTEM	11	06.00	04/01/83
WRMR	WRITEMASTER WORD PROCESSING SYSTEM	9	00.67	01/09/84

32K Classroom

32K Classroom is a regular column aimed at explaining various programming techniques using 32K Structured BASIC. Users are encouraged to submit examples of their own which may help others in understanding and using this powerful language. Call or write I/O News, c/o 32K Classroom, for details on submitting editorial material.

TecEd Note:

The materials presented in this installment of 32K Classroom were found in the Cromemco Customer Support Software Report, dated October 12, 1982. Those items headlined as BASIC apply to both 16K BASIC and 32K SBASIC; those headlined as SBASIC apply only to 32K Structured BASIC.

BASIC-1

Subject: Use of Control P and $LP or $75

Page 56 of the SBASIC Sep 79 manual and page 66 of the 16K BASIC Dec 79 manual refer the user to a handy technique for printing in a BASIC-CDOS environment by using ^P (Control P).

Several clarifications are vital to caution the user about using this in his software. Activating the system printer via a control P is NOT!! the same as using the $LP BASIC printer driver. ^P uses the tty driver to send output to the printer as well as to the terminal. Thus the driver is not a true printer driver and will behave strangely with certain control and escape characters ordinarily sent only to printers. For example, a ^T S for condensed characters on a 3703 or 3715 will lock up the terminal and cause no printed output when used via the ^P feature. Opening $LP works correctly, however. Likewise certain control codes masked off by a tty driver may not get to the printer as desired. It is much better programming practice to use the line printer driver provided in BASIC ($LP for parallel printers and $T54 for serial). This not only guarantees correct operation of the printer, since it is using its own driver, but also allows you to direct output selectively to the printer and console and not just rely on the ^P toggle to send info to the printer.

A final convincing argument against use of ^P, is the fact that using $LP guarantees upward compatibility of your software from CDOS to CROMIX. Output sent to $LP in BASIC responds to the same in CROMIX as it does in CDOS. ^P, ^T, and ^W are not implemented in the CROMIX CDOS Simulator, so that software written with these toggles will not work.

Actual use of the ^P feature is discouraged except for limited use during development and debugging in CDOS.

BASIC-2 Subject: Access to printers

Under CROMIX there is a very convenient way to gain access to several different printers in BASIC.

First make a link from the desired printer to the current directory from which the BASIC program will be running. Within the BASIC program you can open a regular file, then access it through the opened channel.

As an example, suppose a serial printer, SLPT3, was to be used by the BASIC program. First

```
% maklink -v / dev /slpt3 / basicdir
dir
```

Then in BASIC

```
>> OPEN\1,128,2\"slpt3"
>> PRINT\1\ "this is a test"
```

There are two things to watch for. First the printer file must be opened with WRITE-ONLY access (the '2' in the OPEN statement). This will keep you from getting channel access errors. Secondly, it is important to be aware of the fact that you are directing output straight at the printer and not spooling. In a multi-user environment good programming practice dictates that you always use the spooler for output to the printers. In such a case the BASIC-CROMIX command line patch [see I/O News, Vol. 2, Number 2, pp 24-25] will allow you to incorporate spooling instructions within a BASIC program. The technique described here is good for simple cases and applications where you have taken printer collisions well into account.

BASIC-3

Subject: Block device use in BASIC programming

The same technique which was applied to making a link to a printer device and then opening it as a regular file to write to in a BASIC program also applies to block devices such as floppy disks and hard disks.

For example, if we maklink /dev/fdb to the current directory of a BASIC program we can then open this file for read and write.

```
10 Open\3,128\"fdb"
20 Print\3\"This is a test"
30 Print\3,5\"CROMIX is ready"
40 Get\3\a$
50 Get\3,7\b$
60 End
```

Note line 20 above will print "This is a test" starting at the very first byte of the disk residing in FDB. Line 30 causes the line "CROMIX is ready" to be printed at the start of the fifth logical record on the same disk. Note that we have the capability to write to a precise point anywhere on the disk including the system area. This is extremely useful and allows the development of some very powerful disk repair type of application programs. Lines 40 and 50 exhibit the same capabilities in a read. Note that while such disk access power can be very dangerous, there is a level of security already built in due to the fact that the block devices such as "FDB" are owned by system and are therefore inaccessible to someone who would try to use such a scheme in a non-privileged mode. A neat application for this could be to use a disk totally for data storage and not put a file system on at all. Simply writing to the disk in a raw fashion to store data for a BASIC program saves all the overhead used by the directory superblock structure. Naturally only another such program could then access the data so written.

SBASIC-1

Subject: BASICGEN

BASICGEN must be invoked with the name of the (expected) resultant .com file on the command line. Example:

% basicgen mybasic.com

If the file name is omitted, or it is too long (8 characters plus extension) or the extension is not '.com', the error Message:

ERROR: Bad or missing file name.

will be displayed, and BASICGEN will abort.

SBASIC-2

Subject: Sys(14)

Sys(14) is used to determine whether the console driver (either $CO or $SY) is in line or character mode. The default setting is 0 (line mode). You may use the SET command to change it to 1 (character mode).

Character mode is necessary under the CROMIX operating system to use functions such as NOECHO and timed input. For all other uses, the default setting should be used.

SBASIC-3

Subject: Control Z and PEEK

The Control Z (close file) character, when issued from a BASIC prompt dumps you back into the operating system. This is unexpected and could be dangerous.

One can't PEEK at −32768 even though this is a legal value. SBASIC returns with an error 202 (Function argument value).

```
Print PEEK (%8000%) works fine
HEX$ (−32768) works fine
but
Print PEEK (−32768) fails.
```

PROBLEM WITH SAVEing BASIC PROGRAMS

One problem that is encountered time and again by those learning to use 32K SBASIC results from SAVEing programs after having executed them and then modifying them. This seems to manifest itself by not retaining the most recent changes made to the program, or that subsequent execution of the program results in meaningless errors. This is apparently a problem with the SBASIC interpreter. The problem can be even more pronounced when loading and saving a program using a disk drive specifier, such as B: (translated to /B under CROMIX).

There is a way around this problem, although it is rather inconvenient. The technique is to first LIST the modified program to the disk, preferably giving it an extension that flags it as being a LIST version, then executing the SCR (scratch) command to clear memory, ENTERing the LISTed version of the program, and then SAVEing it (with a different or no extension to flag it as a SAVEd version). For example, suppose that you have just executed the program MYPROG, and found and fixed some bugs. The procedure used to guarantee that your changes will be reflected in the program is as follows:

```
>> list "myprog.lst"
>> scr
>>enter "myprog.lst"
>> save "myprog"
```

Although a bit roundabout and time consuming, you can then rest assured that the changes made to the program will be saved.

Inside CROMIX

William E. Jaenicke is an independent software consultant and president of SASi (Satellite Accounting Systems, inc.). In addition to the SASi General Ledger, his firm also developed QTS, a time-keeping and time management report package for professionals. He has been working with Cromemco systems for more than four years, including almost three years of experience with CROMIX. Jaenicke holds regular monthly seminars on CROMIX in his Newport Beach, California offices. He can be reached by phone at (714) 955-2220.

Editor's Note:
User's are encouraged to submit their findings regarding the working of CROMIX so that they may appear in INSIDE CROMIX for the benefit of all. This will ensure that there will always be fresh and informative material. In this spirit, we have received two worthy contributions.

PART I
A COMMAND FILE TO MANAGE MIDNIGHT DATA BACKUP

by Leigh Thomas

Despite the fact that we all know the importance of keeping backups of our valuable data files, it is genuinely amazing how often we all fall into the trap of laziness, on the assumption that the system will keep running and that a backup can wait till later. In short, most of us put off till tomorrow what we should do today.

Part of the trouble is that making backups is often a nuisance, especially for those sites without tape systems. Fortunately, command files can come to the rescue and this article describes one such file which will run automatically every night without user intervention. All you must do is remember to put a disk in.

In concept, the example is simple though some of the detail is a little complex.

The command file contains three distinct sections—

```
The command file contains three distinct sections -
     wait
          is it midnight yet?
          if NO, go back to wait
          if YES, go on to backup routine
     backup
          (assumes a CROMIX floppy is physically in drive)
          mount the floppy
          was there an error when trying to mount?
          if YES, goto error section
          if NO, continue from here with backup
          invoke a command file to do any "find"
          that may be necessary - see below
          copy all necessary files to floppy
          write the date and time to the floppy
          unmount the floppy
          clean up any temporary directory entries
          created by the find command file - see below
          go back to wait
     error
          write a message on the system terminal
          saying that an error occurred and that
          backup did not take place.
          go back to wait
```

The listing of the command file to achieve this is shown.

DESCRIPTION OF THE "BACKUP" SECTION

This section actually does the work, and comes into operation sometime shortly after midnight. See the description in a subsequent section about this waiting process.

The files to backup may be nominated explicitly in the command file using their full path names or it may be more suitable to backup all files which are new or changed today. If the latter applies to your needs it is obviously impossible to know in advance the names of all these files when you write and run this example backup program. You could, however, use a general statement in the backup section of the program which said something like—

```
copy -f /usr/backup/** /b
```

which assumes that all files to be backed up in the nominated way exist (either as copies or as maklinks) in the directory created for the purpose, named/usr/backup in the above example. Of course, all new or changed files won't be there unless another command file finds them and deliberately puts them there, but this is simply attended to by a one line command file

based upon the CROMIX FIND command. For example,

```
name = scan.cmd (in the /cmd directory)
find /usr -type f -a -mtime -1 -a -exec
maklink -f{ } /usr/ backup
```

In plain English, this means—

"FIND, starting from the /usr directory and going down from there as far as necessary, all entities of the type "file" (the f option) and (the -a part) which have been modified (or created) one day or less ago (the -1 option) and (the -a again), when you find these things, execute (the -exec) the CROMIX instruction to the right of the "exec" part. This CROMIX instruction is simply the familiar MAKLINK, but the file name to link is replaced by the curly bracket pair { } which is part of the syntax of the FIND instruction."

Thus, each file found will be new or newly modified and will be linked into the directory /usr/backup which you must separately create for this purpose.

The backup command file needs only to contain the instruction—

```
scan
```

in the backup section to collect all necessary files, and then may be simply followed with a line like—

```
copy -fv /usr/backup/** /b
(or whatever, as required)
```

If you follow this procedure you would want to clean up the contents of the /usr/backup directory once its purpose has been served so that it is emptied, ready for use again tomorrow.

Thus, your backup command file will probably contain a line like

```
del /usr/backup/** >* /dev/null
```

which says, delete all contents of the directory we used to collect the files to backup and redirect any harmless errors to the null device (garbage bin).

After completing the backup procedure by copying to the mounted floppy, the disk is unmounted and control reverts to %wait, where the file will pause until next midnight. If desired, the time function may be used as shown to redirect the time and date of this backup to a file on the floppy before unmounting.

The mount instruction, showing drive "b" in this case, is followed by the line—

```
if -err goto error
```

which says, if the mount fails, transfer control to the section of the command file named %error. The relevant section simply types a pre-written message, which is stored in a file in the /cmd directory called "faultmessage," to the system by virtue of an output redirection. Thus it does not matter which terminal originally invoked this backup process; the error message, should it ever occur, goes only to the console.

After sending the message, the command file goes back to the waiting phase.

DESCRIPTION OF THE "WAIT" SECTION

This is the most complex of the three sections, utilizing the CROMIX features of DAY, CMPASC, INPUT, TESTINP, error trapping, sleeping and redirection.

The very first time the process begins, the DAY function creates and writes into a file in the /cmd directory called d2.

The DAY utility simply generates a message saying what day it is today, as determined by the system clock. Typing DAY at your terminal will reveal a message like—

```
Today is Tuesday
```

or whatever. The output is directed, not to the screen in this case, but to the file d2.

Immediately thereafter, exactly the same thing is performed again, but this time the message is directed to a different file, named /cmd/d1.

Next, the CMPASC (compare ascii) utility is invoked to compare the contents of d1 and d2. The output of this comparison process will be a message telling us whether or not they do compare. It would normally go to the screen, but an output redirection places it into a new file, named d3 in the /cmd directory.

If the two files are identical, the result will be the message—

```
/cmd/d1 matches /cmd/d2 exactly
```

which is now in file d3, where it may be read by the command file.

If the two files are not the same, the output of the CMPASC function is quite different. It is then a multi-line output highlighting the differences, and looks like this—

```
----> /cmd/d1
Today is Monday.
----> /cmd/d2
Today is Tuesday
```

For the purposes of the "wait" section of the command file under discussion the exact format differences are immaterial except insofar as they give us a way of simply determining whether or not the two files are the same. If the files are identical it is not yet midnight. At or slightly after midnight, one file will say "Today is Thursday" and the other will say "Today is Wednesday" or similar, as there is a time lag between when these two files are normally - written as is described below.

The input function is used with two redirections, one an input, one an output. The input redirection causes the input function to lift off the first line of the nominated file (d3 in this case). The output redirection tells the function what it must do with the extracted line, put it temporarily into d1 in this case, which destroys what was originally in d1. This doesn't matter as we shall see.

The next line, TESTINP, examines the contents of file d1, the -f option telling TESTINP only to bother with the first character in the file. The slash (/) following the file name tells TESTINP which character it must look for. As you can see from the example above, if the files are identical then the line saying this begins with a "/" character. If they are not the same the first line of the output saying so does not begin with a slash. Thus, we have a ready-made method of determining a day match.

If the TESTINP function did not find the slash, the days are different. It is therefore roughly midnight and the backup should go ahead. Branching to the backup section then takes place.

Most of the time it is not midnight (obviously) so the usual result of the test will be that the days are the same, so file d1 will start with a slash. When this happens, it is not yet time to do the backup.

The command file then involves "day" again, putting it freshly into d2, and then sleeps for half an hour, during which time nothing happens. Upon reawakening, the command file continues operation from where it was before sleeping and encounters a goto instruction which transfers control to "wait." The first thing wait does is put the day into d1 again (thereby overwriting the temporary data put there previously during the testing phase) and the whole process happens again.

SUMMARY

In conclusion, this type of command file set offers a reasonably flexible and painless way of automatically backing up data on a regular basis, at a time when users will generally not be on the system. It is based of course on the assumption that the floppy has sufficient room to accommodate the growing data, something that you, as systems supervisor, will still need to keep a watchful eye on.

To actually start the command file going you must, as a privileged user (mount is a privileged system call) type—

```
nightsave &
```

where the trailing ampersand puts the process in background mode, where it

will run forever. Alternatively, you may care to put this instruction in your startup.cmd file so that nightsave is invoked at system boot automatically.

```
name = nightsave.cmd   (in the /cmd directory)


day ) /cmd/d2
%
%
%wait
%
%******************** this is the wait routine ******************
%
day ) /cmd/d1
cmpasc /cmd/d1 /cmd/d2 ) /cmd/d3
input ( /cmd/d3 ) /cmd/d1
testinp -f /cmd/d1 /
if -err goto backup
day ) /cmd/d2
sleep 1800
%
goto wait
%******************** this is the backup section ****************
%backup
%
% save day in d2 for next loop
day ) /cmd/d2
%
mount sfdb /b
% (mount returns an error if it cannot proceed for some reason)
if -err goto error
%
% (save whatever is required now .... )
%
scan
copy -f /usr/backup/** /b
%
% (now copy anything else you want as well .... )
copy -f anyfile /b
%
% (now delete files you saved from the temporary backup area)
del /usr/backup/** )* /dev/null
%
% (save the current date and time in a file with the backups)
time ) /b/when
%
unmount sfdb
%
goto wait
%
%******************** this is the error section *****************
%error
% (get here if disk didn't mount)
ty /cmd/faultmessage ) /dev/console
goto wait
%
% ***************************** end ****************************
```

PART II
AN ALTERNATIVE SCHEME FOR BACKGROUND MAINTENANCE ROUTINES

Editor's Note

The following was submitted by Kirk C. Aune, Ph.D., Associate Professor of Bio-chemistry at the Baylor College of Medicine, Houston, Texas.)

I read with interest your discussion of the use of testinp.bin in connection with your desire to perform background tasks at a specified time. Although I have not encountered the problem you described, it could have happened at some point and so I thank you for pointing out the idiosyncracy.

I am writing specifically to offer you an alternative to the scheme you described for maintenance routines in the background. I have used these routines quite successfully. The scheme consists of three programs: 1.) SLP_TASK.CMD, a command file which is the master command file that dictates the tasks to be performed; 2.) CMPTIME.COM, a load module from a FORTRAN source that reads a file named TASKTIME and then computes the time from the current time to the "TASKTIME" and then generates a command file named SLPTIME.CMD. 3.) SLPTIME.CMD, a generated command file that sleeps away the prescribed time in increments of 3600 seconds for the hours and the residual number of seconds.

The file TASKTIME contains the ASCII time as hh:mm:ss with a carriage return delimiter. (Hence, if it is generated within SLP_TASK.CMD it should be generated, for example by: ▶

The reader will note that SLP_TASK.CMD is a tight loop. The process sits in the shell executing once every time the program SLPTIME.CMD times out. The automated tasks could contain additional calls to CMPTIME.COM after TASKTIME is changed as in the above example to perform more than one cycle every 24 hours.

The following is a listing of the FORTRAN program that yields CMPTIME.COM: ▶

The above program is by no means beautiful, but it does its job. It requires the subroutine TIMEX which I use to obtain system time within FORTRAN programs. The Z80 assembler listing is given below:

```
;////////////////////////////////////////////////////////////
;THIS GETS THE CROMIX TIME
        ENTRY TIMEX
;
; This passes year,mon,day,hour,min,sec
; data as an integer*1 array of size 6: T(6)
;     FORTRAN         CALL TIMEX(T)
; Uses CROMIX CALLS
;
TIMEX:  PUSH    BC
        PUSH    HL
        POP     BC
        PUSH    HL
        PUSH    DE
        JSYS    30H             ;GET DATE
        LD      A,E
        LD      (BC),A          ; SAVE YEAR
        INC     BC
        LD      A,H
        LD      (BC),A          ; SAVE MONTH
        INC     BC
        LD      A,L
        LD      (BC),A          ; SAVE DAY OF MONTH
        JSYS    32H             ;GET TIME
        INC     BC
        LD      A,E
        LD      (BC),A          ; SAVE HOUR
        INC     BC
        LD      A,H
        LD      (BC),A          ; SAVE MINUTE
        INC     BC
        LD      A,L
        LD      (BC),A          ; SAVE SECOND
        POP     DE
        POP     HL
        POP     BC
        RET
        END
```

The generated product of CMPTIME.COM looks like the following:

```
% Termination time: 20:10:0
repeat      5 sleep 3600
sleep       350
exit
```

Note that TASKTIME had contained the value of 20:10:00 and the program had been executed a few minutes after 15:00. The comment line allows for an inspection of when to expect the SLPTIME.CMD to time out. Note also that the repeat feature is used for units of hours. Unfortunately, one must use that feature to span a 24 hour time period because of the 65536 seconds sleep limitation. It is not a major problem be-

```
echo 'hh:mm:ss' > /etc/autotask/TASKTIME
ty cr_lf >> /etc/autotask/TASKTIME

where cr_lf is a file with 0dH 0aH forcing a 0DH into the file.)

The form of the program SLP_TASK.CMD is:

%//////////////////////////////
%
%SLP_TASK.CMD
%
% 11/30/83
% kca
%//////////////////////////////
%again
d /etc/autotask
/etc/autotask/CMPTIME.COM
/etc/autotask/SLPTIME.CMD
% ************************
% ** Insert automated tasks to be done here:
% ************************
goto again
```

```
      PROGRAM CMPTME
C//////////////////////////////////
C
C  Computes the repeat sleep times for
C  a specified time.
C  Output is written to a file named: SLPTIME.CMD
C
C  KCA 03/30/83
C
C//////////////////////////////////
C
      INTEGER*1 CURTIM(6),QUTTIM(6)
      INTEGER*1 DUM
C
      INTEGER RPT,SECS
C
      CALL OPEN(6,'TASKTIME   ',0)
      READ(6,100,END=1) QUTTIM(4),DUM,QUTTIM(5),DUM,QUTTIM(6)
  100 FORMAT(1I2,1A1,1I2,1A1,1I2)
    1 ENDFILE 6
      CALL TIMEX(CURTIM)
      TC=CURTIM(4)+CURTIM(5)/60.+CURTIM(6)/3600.
      TQ=QUTTIM(4)+QUTTIM(5)/60.+QUTTIM(6)/3600.
      HOURS=TQ-TC
      IF(HOURS.LT.0) HOURS=HOURS+24.
      RPT=INT(HOURS)
      SECS=3600*(HOURS-RPT)
      CALL OPEN(6,'SLPTIME CMD',0)
      WRITE(6,101) QUTTIM(4),DUM,QUTTIM(5),DUM,QUTTIM(6)
  101 FORMAT(' ','% Termination time: ',1I2,1A1,1I2,1A1,1I2,/)
      IF(RPT.GT.0) WRITE(6,102) RPT
  102 FORMAT('+','repeat ',1I5,' sleep 3600',/)
      IF(SECS.GT.0) WRITE(6,103) SECS
  103 FORMAT('+','sleep ',1I5,/)
      WRITE(6,104)
  104 FORMAT('+','exit',/)
      ENDFILE 6
      END
```

cause it is executing every hour in the shell rather than in a bank of user memory.

I have created a directory name autotask which is sub to /etc to hold the files used in the process. That is at the discretion of users. The items that are important are that SLP__TASK.CMD can find CMPTIME.COM, CMPTIME.COM can find the file TASKTIME in the current directory and SLPTIME.CMD times out to the directory where the tasks are to be performed.

The whole process is started by executing SLP__TASK.CMD as a detached job. I frequently have it executed as a line in the startup.cmd and is hence, silent to users:

```
d /etc/autotask
slp__task >* /dev/null&
```

That places the process in the proper directory and dumps all verbage that could result out to the black hole.

The automated task may be started prematurely by killing the PID associated with SLPTIME.CMD:

```
kill -3 PID
```

It is self-restoring. If it is desired to terminate the process, one should FIRST kill -3 PID for SLP__TASK.CMD and THEN kill -3 PID for SLPTIME.CMD.

One can add additional features such that the SLPTIME.CMD can time out and then have SLP__TASK.CMD continue to check every five minutes until who.bin shows no person logged in. UNDER NO CIRCUMSTANCES would one want this background task to begin execution where disk activity was involved if other tasks such as disk directory and inode checks, cptree, and diskcopy routines were occurring.

I hope you find this sufficiently useful to pass along.

Kirk C. Aune, Ph.D.

tec·tips

TEC TIPS is a regular column aimed at providing hints for keeping systems up and running. It will not attempt to deal with specific engineering applications or non-standard configurations. TEC TIPS is edited by Richard Quinn, owner of QUINTEC, a Southern California Computer service firm.

Correction from Cromemco

An item run in this column in Vol. III, No. 6 indicated that Cromemco was replacing revision E, G, G1, or older 64KZ cards with 64-KZ-2 cards. This was in error. Cromemco's policy is to replace, rather than repair, older 64KZ cards. But the replacement card is a newer version 64KZ card, not a 64KZ-2 card.

CHEAP ANTI-STATIC CURE

As most computer users know, static is an enemy of electronic micro chips. It can cause damage that results in instant failures, or does not result in a failure until some time later. There are several ways to help.

When new carpet is installed in your office, or you are renting an office for the first time, be certain that the carpet is anti-static treated. This is the longest lasting and best treatment of all. Special computer carpets even have carbon particles in the fibers to stop static, and provide the best prevention.

There are a variety of anti-static pads and mats for computer work stations. It has been my experience that these are expensive and rarely used.

What's my solution? First I make certain that all outlet grounds are good and that all equipment is plugged into good three-prong outlets. I check between the hot and ground on all outlets to be certain that there is 120 volts AC, and then I check between the neutral and ground to be certain that there is 0 volts AC. If these conditions are correct then most static dissipates through the metal case to ground with much less harm.

The second thing I do for my customers is to take one cup of Downey fabric softener in a gallon of water and put it into a Hudson type pump sprayer: the type that you can buy at the garden store or lumber yard. I use the plastic version so rust will not get in the system. I then spray the carpet wall to wall with the solution: no more static! It works great and costs only a few dollars, compared to the more expensive sprays in a can. You may have to repeat the treatment from time to time for best results, but that is true of most treatments.

Preventive Maintenance

What preventive maintenance is needed by my system? How can I keep it running at peak reliability? These are questions often asked by users.

There is really very little preventive maintenance that can be done. What can be done revolves around dust and dirt. The amount of this that needs to be done is directly related to the amount of dirt in your area.

Keep all fan grills clean and free flowing. You can buy an aerosol can blower at most TV or electronic supply stores. Don't buy those with any kind of oil or lube: this will attract dirt and make the problem worse. Keep the grill clear to keep the air flowing. When we pull a system in for service, we use dry compressed air to blow the dirt out of everything.

Use denatured alcohol, the type purchased from paint and hardware stores, to clean the belts on the floppy drives. If yours is a floppy-only system, clean them every 6 to 10 months. If you use a hard disk you might get by without a need for cleaning for several years. Check the RPM when you initialize your diskettes (later versions of INIT report the RPM) to see if the drive is running at the proper speed. Five-inch floppy drives run at 300 RPM and eight-inch floppy drives run at 360 RPM. Other things can cause RPM to be off, so if cleaning the belts doesn't fix the problem, have the drive checked.

Some have asked about an air filter. You can get an air conditioner filter pad and cut it to fit over the opening if the fan on your system is drawing air in from the back, in the case of a CS-2 and a CS-3, or the side, in the case of a CS-1. If the air is discharging from the fan, don't put a filter over it as this will restrict air flow, and thereby cause heat related problems.

The CS-3A draws air from the right side and the bottom, so a pad is harder to install over all openings. The problem with a pad is two-fold; if the pad is too dense or small it will slow down air flow (a great no-no). In addition, if the pad fills with dust or dirt and is not cleaned often, it likewise will restrict air flow, with the same damaging results.

I install all fans in CS-2s and CS-3s with the discharge to the inside of the cabinet. I find this pushes the hot air out best, and the air can lose most dust or dirt before it reaches the disk drives in the front. Blowing air into the cabinet gives the dust a chance to settle in the back of the cabinet (in the power supply area where there are no moving parts) making a filter unnecessary in most installations.

Because of the openings around the drives, the air leaks out, which helps cool them as well. If the fan is drawing air out of the cabinet, the dirty air is drawn first through the drives and grills, loading them with dirt and causing disk wear. If you have a hot room, a larger fan (120-160 cfm) can be purchased that will fit in the same opening. It may be noisier but it will cool better.

The newer CS-3As and all versions of the CS-1s and CS-1Hs have plenty of cooling and work fine as equipped from the factory. Older CS-3s and CS-2s may need to have the fans turned around, or retrofitted with larger ones. I like to keep the interior of the cabinet under 90 degrees Fahrenheit, or so. The cooler the better, as long as no condensation occurs.

Keeping a Journal on Your System

Without a doubt, the most frustrating thing I deal with in trying to help people is related to "what happened when I did what." I often get calls where users tell me that their computer "just quit" only to find out through questioning that many events led up to the "just quit." They are worried I will accuse them of causing the problems through something they did, which may be true, but it still needs to be known to solve and prevent the problem.

I recommend a journal or log for each system. This is especially important for CROMIX systems or systems used by several people. If good careful records are kept you can often trace those rare failures to their sources.

Print the form sideways on an 8½ by 11 or legal 8½ by 14 to give the most space for written comments. (Yes, you will have to write it by hand because if you keep it on the system you may not have it when the system goes down!).

What kind of things do I keep in the log book? Well, the obvious things like date and time of entry, (time is important, as I know of several systems that failed at night or early morning when time clocks killed air conditioners or power companies switched mains), the person using the system (some users have a special talent!), the program running in case the problem is software

related, the problem reported and what action was taken (no screaming, foul language or tears in the log, please). You don't know how useful a log can be till you try to track an elusive problem that only occurs every leap year.

What's most important is to enter entire error messages when the system reports a problem. All those numbers and words mean something, even if you don't know what it is. An experienced operator or serviceman can make a great deal of sense out of them, if you keep accurate notes. I would say that every error message should be entered into the log, unless you know what it was from, and that it was only an isolated event.

Date/Time	User ID	Program in Use	Problem Reported	Action Taken

The log will show patterns that will lead you to solve hardware and software problems. In addition, it is the best place to quickly find information regarding software version changes or updates put on your system, the last backup of the hard disk, when that new terminal was added, when the memory was tested and so forth. Try it. You'll find it very useful and so will your serviceman.

Close Encounters of the C-10 Kind

Close Encounters of the C-10 Kind is a regular column directed to users of Cromemco's personal computer, the C-10. It is edited by Dr. Tom Beer, of Applied Environmetrics, located at 1418 Gordon St., Balwyn, Victoria 3103, Australia. Dr. Beer can be reached by phone during business hours at 8180264, and at home at 8022571.

"Och Would some power the giftie gie us, to see ourselves as others see us" (Robbie Burns, Scots poet)

I like to spend the occasional few minutes browsing through the magazine stands of our local news agent. Over the past year the space devoted to computer magazines has increased at an amazing rate and I flick over many of these looking for articles that are either about the C-10, or would be useful to C-10 owners. So far I have only come across three articles directly on the C-10 itself, and it may interest other C-10 owners who have not seen these particular articles to have a brief resume of them.

The first article that I saw about the C-10 appeared in print just a little bit after I actually got my machine. It appeared in **Electronics Australia** (August 1983) under the byline "Professional Value for Money" and was written by Peter Vernon. He was quite complimentary about keyboard, terminal, and the system configuration and was definitely enraptured with SBASIC ('It combines the clarity made possible by structured program design with the ease of use of a BASIC interpreter'), with WriteMaster ('an excellent program') and with PlanMaster ('flexible and convenient to use'). Vernon's gripes were as follows: the keyboard beep was too piercing and too long and the keys have a heavy action. The power switch is in an awkward position and there are no brightness or contrast controls on the terminal. He was also puzzled by the inability of CROS to set breakpoints or jump Z80 registers, both of which are essential for debugging machine language programs.

In fact, my understanding of CROS is that it was never intended as a machine language program debugger (Cromemco supplies the program DEBUG to do that) but acts merely to control the C-10 immediately after the power is switched on. Most of the time the main function of CROS is to load CDOS, the disk operating system.

Given the date of Vernon's article he would not have had too much time to get to grips with his C-10. Martin Marshall writing in the Sept./Oct. 1983 issue of **Small Business Computers** would also not have had too much time with his machine, but his experiences made my hair stand on end and my teeth drop out. He had two disk drives fail on him and found that the C-10 screen stability was temperature-dependent. One of the chips had to be warmed up on a cold day for 15 minutes before it would produce a stable display. Just for the record, my disk drive has now performed flawlessly for a year and the screen comes on at about the same time that CDOS loads. Given that Australia's climate and San Francisco's are not too dissimilar, it certainly looks as if poor Martin Marshall was supplied with a very early version of the C-10. In fact, Marshall was reviewing Word Processor programs and intercompared the Osborne, Kaypro, Morrow and Cromemco offerings. WriteMaster got some harsh knocks, some of which must have been taken to heart by Cromemco for I notice that Release 4 WriteMaster now displays the cursor location—an omission from earlier releases that Marshall found most distressing.

Another of Marshall's criticisms was a bit unfair (as, no doubt, are some of mine) as it could have been overcome by a bit of knowledge about the way the system worked. Marshall was scandalized that it took 69 seconds to actually start writing something once you turn the machine on. The delay arose because after the switching on, CROS loads in CDOS, then CDOS loads in MENU, then after making the appropriate choice WriteMaster is loaded into memory. Whew! Even with the early release software this process could be speeded up by using the Batch utility. With present software a file called STARTUP.CMD can be created which will immediately load WriteMaster instead of the Menu. This works fine, though for some unknown reason STARTUP.CMD and SBASIC.com often do not get along well together.

I actually agree philosophically with Marshall's idea of time saving. He calls it psychonomics. One time wasting feature that intensely aggravates me is the necessity to load COPYFILE for each file that you wish to copy. A similar wave of frustration sweeps over me when I have been working in BASIC, have said BYE, returned to CDOS and then realize that I want to return to BASIC to do something else. The frustration arises because I know that in both cases the program that I want is still sitting in memory, and the act of commanding CDOS to the program will take time and reload a program that is already there. There is a very neat solution, however. On every disk I have a file called RERUN.COM that has nothing in it. It is produced by calling up BASIC and issuing the command CREATE "RERUN.COM" and then issuing the command BYE. This produces a blank file. If I then wish to rerun a program that is already loaded in memory I merely have to say RERUN. Then CDOS loads the blank program (i.e. the memory remains unchanged) and starts to re-execute whatever the previously run program was. CDOS can load a program of 0K much faster than one of 26K, and I found that RERUN will get me back into BASIC in 4 seconds compared to the normal 12 seconds that it takes CDOS to load SBASIC.

Though Marshall finds WriteMaster 'perfectly usable' he was happy neither with the C-10 screen (because the bottom of some characters overlap with the top of others on the line below; as with the y above an h) nor with the arrow key arrangement on the keyboard. On this last point I do tend to agree that the diamond shaped key arrangement on some keyboards, in which key location corresponds to the arrow function, does seem to make sense. Finally, Cromemco insularity is noted. Cromemco disk drives, at that time, could only read Cromemco diskettes. The very recent release of DiskMaster will hopefully have partly resolved that problem.

Lastly, the January 1984 issue of **Creative Computing** had an in-depth evaluation of the C-10SP by David Hilton, under the byline "A reasonably priced Business System." This is a very well written and well thought out review

that was based on three months usage of the C-10. Obviously Hilton too must have had problems because in his three months of operation he was given four releases of the operating system, and two different computers! The reason? He was getting all sorts of strange results from WriteMaster and PlanMaster in which the system would hang, return him to CDOS or the MENU, garble a display or destroy his file. He explicitly points out that neither he nor the people at Cromemco know just how many of these problems are bugs in the software and how many are attributable to hardware failures. Whatever their cause, some of them are still there. I wrote this article using Release 4 WriteMaster and had the whole screen flashing at me after executing a Jump to End command, whereas another time the whole system hung after I held down the TAB key too long.

Hilton was happy with most of the hardware, except for the keyboard height adjusters (the two rubber-tipped bolts underneath the keyboard) which he felt were inadequate. Most of his review dealt with the software and his ranking order is really most intersting. COPYFILE, SCREEN, and SBASIC really seem to have won his heart, as did WriteMaster. To quote "WriteMaster is almost reason enough to buy the machine." At the other end of the spectrum however, Hilton did not have kind words for Planmaster which, according to him, fell far short of the quality of the rest of the system. He suspects that anyone who is serious about using an electronic spread sheet will want to look for a different program.

Hilton's single most damning question is: Why do the Master programs jump outside of themselves so often and end up in the System Trap? Apparently part of the boot procedure sets the contents of RAM to EF hexadecimal which is the Z80 opcode to Restart at memory location 28H. The Technical Manual tells us that 28H is the 'Jump to System Trap Message' and the suspicion seems to be (I think) that there are stray EF's lying all around the innards of memory, and when a poor unsuspecting program bumps into one of them it is kicked off unceremoniously to 28H. Unfortunately the Technical Manual does not tell us any more about the System Trap message. What, of course, is extraordinarily frustrating about the occurrence of these jumps to the System Trap is their irreproducibility. Random, unpredictable behavior is not supposed to be characteristic of computers.

Until I read Hilton's article, I had felt that deep down I must have been to blame whenever the system hung. Now I feel that I am never to blame and it is all the fault of the machine. The truth is

probably somewhere in between, and the cure for all these ills is still relatively simple. Turn off the machine, count to five, turn it on again and it will reboot and unhang. However, I still wonder who or what was responsible for my corrupted disk files (gremlins??). After about three months of usage the Chess program on the System disk hung and refused to work properly. I eventually had to delete it and restore CHESS.COM from the backup disk. Seven months later SBASIC was corrupted. Is this just par for the course when one uses floppies or is this a subtle machine-disk conspiracy? I do not know, but readers' views and reactions will be welcome.

Dr. John Parrish of Emporia State University celebrated the first anniversary of his C-10 ownership by writing to I/O News and detailing some of his interesting experiences and discoveries. Before handing over to him I should, however, explain about DEBUG.COM, which he mentions in his letter. As this is a Cromemco utility that is not supplied with the C-10, but must be purchased separately, many C-10 users may not have a clear understanding of its function.

DEBUG has four major functions. Firstly, it allows one to dump each byte of a file and to change the byte as so desired. Secondly, it will trace through user programs and is thus an invaluable aid to anyone writing assembly language programs. Thirdly, it can be used to list the contents of a program file in assembly language mnemonics, a process commonly called disassembly. And fourthly, DEBUG enables the user to program PROMs (Programmable Read Only Memory chips). When DEBUG is run it adds 100 (Hexadecimal) to each byte number, so that the address 09B7 (Hexadecimal) is actually byte number 08B7H (the H at the end indicates that it is a hexadecimal number).

Dr. Parrish has utilized the first of these in order to alter the PRINTER.COM file. For those of you with innate programming skills and a miserly bent, the dumping and changing of bytes can be done with SBASIC. For example, to make the first of the changes specified by Dr. Parrish this SBASIC program will do:

```
 5 Open\2\"printer.com"
10 Dim Existing'text$(13),New'text$(13)
15 New'text$ = "Okidata u-92 "
20 Get\2,0,%08B7%\Existing'text$
25 Print Existing'text$
30 Put\2,0,%08B7%\New'text$(0,13)
40 Close
```

Run it twice to check that the change has been effected.

Enough from me. I now hand over to Dr. Parrish who begins with some wise words about the menu.

The Menu

The MENU.DOC program is exceedingly easy to modify for personal uses. Since it is written in SCREEN format, all one has to do is to remove all of the attributes from the file (attr menu.doc), load it into SCREEN (screen, menu.doc), and make any of the desired changes. Save those changes and load in the "new" MENU (be sure to re-attribute the file when you are through editing it: attr MENU.DOC EWS). Note that all of the programs that you want the Menu to run must be preceded by an asterisk (*). Thus, *printer 8 as one of the Menu programs will cause the printer program (PRINTER.COM) to select a parallel printer when one either presses the number corresponding to that file or types in the program's name (Printer 8) on the Menu's prompt line.

A note of CAUTION: when you delete a file to make room for one of your own creation, make sure that you maintain the same number of files in your "new" MENU.DOC (19 files).

It also is possible to have "batch files" (files with the filename.cmd format) in your Menu. I have a parallel dot matrix printer and since the default printer for the C-10 is a serial printer one must load the printer menu, or run the printer.com file, before the C-10 will recognize that a parallel printer is attached. I have a menu program in the number 8 position in my Menu entitled *OKI.CMD which runs the following programs in succession:

```
printer 8
sbasic cqp.sav
writmast
```

The first program loads in my parallel printer, the second calls up 32K Structured BASIC and runs an SBASIC program that puts my Okidata dot matrix printer into its correspondence quality mode:

```
 5 Open\1\"$LP"
10 @ \1\Chr$(27);Chr$(49)
15 Close\1\:Bye
```

and finally, the batch-file loads in WriteMaster. The effect of this batch-file, thus, is to set-up both my printer and my computer for wordprocessing. This makes setting up my system for wordprocessing almost effortless since all I have to do is press the #8 key while in the Menu (alternatively, I can type Oki or Oki.cmd on the Menu prompt line). Batch-files, by the way, also are written with the SCREEN program (e.g., Screen Oki.cmd and put in the programs to call so that they are stacked sequentially).

Debugging Files

Most recently I have been involved in a somewhat more adventuresome form of hacking using the DEBUG.COM program. For example, I have debugged the PRINTER.COM file by replacing the hex-code for "Other parallel" printer with "Okidata u-92" printer (addr:09B7,09C3), and also replaced the hex-code for "parallel printer" with "Okidata printer" (addr:161E,162D). Now when I type Printer 8 on the Menu prompt line, or after a CDOS "A." prompt, the C-10 screen shows "Okidata printer selected," and the Printer.com Menu now shows that an "Okidata u-92 printer" is the option next to printer selection #8. While these changes are not necessary for the printer to be functional, it was gratifying to figure out how to use DEBUG to accomplish those feats.

Another important use of the debugger is to UN-ERASE accidentally-erased files. These can be salvaged as long as you have NOT written any new files to the diskette since the accidental erasure. To correct this problem use the following command: Debug Sys.dir. This will allow you to list the files on the diskette (just type dm and hit RETURN). Those files preceded by E5 are interpreted as being erased. Using the substitute memory command, substitute 00 for the E5 and the file will now be accessible (be sure to write the changes to the file before exiting debug!). DEBUG is a useful utility, but the documentation is "typical" Cromemco, meaning that it is not written for the novice. Seek assistance from an experienced person, if possible, before attempting some of the more sophisticated capabilities of DEBUG.COM (ALWAYS make sure you make changes on copied diskettes since any changes you make cannot be corrected unless you still have the files on your original diskette).

Control-Shift Functions

Hold down the CONTROL and SHIFT keys at the same time and while continuing to hold them down, press the letter S. This will cause your status line to appear on line 25 of your screen (ONLINE or LOCAL I PAGE I DSR I MAIN I READY OR LINE LOCK I PAGE INSERT I GRAPHIC I I). Now if you hold down the control-shift keys again, and press the letter L, you will be in the local mode (note these keys are toggle switches; press them again and they change back to the prior condition). While you are in the local mode you can easily change the type of cursor marker that you have. While still in the local mode, press the ESC and period (.) keys in succession. Then, the third successive key you press (UPPER CASE letter) determines the type of cursor: A = reverse-video cursor (default); B = underline cursor; C = blinking reverse-video cursor; D = blinking underline cursor (e.g., the sequence ESC . D changes the cursor to blinking underline). After selecting the cursor of your choice you must get out of the local mode, which is easily accomplished by pressing the letter L while you hold

down the control-shift keys. You can determine whether you are in local or online mode by looking at the status line again (hold down the control-shift keys and press the S key).

Another function which can be evoked from the local mode is to turn on and start the C-10's clock. Turn on the status line again (Control-Shift S). Get into local mode (Control-Shift L). Now press ESC and then the spacebar, followed by six numbers which represent the time (HHMMSS) in a 24-hour clock. Now toggle-out of the local mode and your clock is always available to read when you toggle-on the status line. Alternatively, you can set your clock to 00:00:00 and when you toggle-back to the ONLINE mode, you will be able to time a session in front of the C-10.

Three last useful commands for the C-10 include one which allows you to use the letter L to list the directory from the Menu or the A. or B. prompts (CDOS), instead of DIR as discussed in the C-10 Users Manual. Another command that can be used from the A. or B. prompt is the command TY instead of TYPE. Both can be used to type-out any ASCII file (e.g., ty oki.cmd or type oki.cmd). The third useful command is the use of the abbreviation DEL (delete) to remove a file from a disk in CDOS. This function obviously complements the ERA (erase) command.

Using WriteMaster or SCREEN, try typing in the following program under the filename: sbasic.bas (once you go to SBASIC, type the following after the prompt: >> enter "sbasic.bas). Once you have entered the program type the following after the prompt:>> run. Here's the program:

```
5  @ Chr$(25);"dH"
10 @ "Type the numbered basic statements";
15 @ " to create your own program."
20 @ "To go back to the C-10 Menu, type: ";
     Chr$(27);"dBbye"; Chr$(27);"d@"
25 Scr
```

If you like what you have done, era sbasic.sav and rename (ren) sbasic.sav = sbasic.bas (note, if you created the above program using WriteMaster, convert the text to Screen before exiting WriteMaster).

Hopefully, the above tips and ideas will make your time in front of the C-10 more enjoyable and rewarding. There are other tricks that can be useful, but I'll let you others out there send them into the I/O News for future articles. Until then, happy C-10ing.

Soft Tips

SOFT TIPS is a regular column aimed at providing software oriented hints and ideas for non-programmers. Members are encouraged to send in tips that can help a user better use his/her system. SOFT TIPS is designed to put forth ideas that are general in nature. The column is edited by Norman Vadnais, President of **Computer Specialists & Associates,** an Orange County Customer Support Specialist. Member's contributions can be sent to SOFT TIPS in care of I/O News. If you wish to discuss your software situation directly with Mr. Vadnais, he can be reached at (714) 841-3620.

LETTING CROMIX RUN YOUR SYSTEM

Letting CROMIX Run Your System has spent much time explaining the proper setting of both users and groups on a CROMIX system and how these proper settings can improve the system. The access privileges of the files on the system very much affect any steps made in this direction. The ability or inability for users to access a file or directory of files is important in setting up any logical user and group system. CROMIX offers four (4) types of access privileges which are maintained for three (3) different sets of users.

The four access privileges allowed are **Read, Execute, Write,** and **Append** (many times referred to as REWA by CROMIX and its users). READ and WRITE access are self explanatory. EXECUTE access allows a program to be loaded in memory and executed by the processor (like a .com or .bin file), and APPEND access allows additions to the end of a file. For example, you need Read access to dump a file, Read and Write access to edit a file, Execute access to run a program, and so on. Which file privileges are necessary when running various application and utility programs, however, are determined by the individual program.

The file accesses of directories are also an important consideration. **READ access allows listing of a directory, Execute access allows the use of the directory in a path name, Write access allows the creation or deletion of files in a directory,** and **Append access seems to have no meaning to directories.** Directory access privileges are determined by looking at all directories starting at the file in question and working all the way back up to the root directory. In other words, you may not enter a directory unless you have Execute access for the entire path-name of that directory. Also note that for CROMIX to determine what wildcards (i.e., *, ?, etc.) are to stand for, it must have Read access for the directory involved.

The combination of directory privileges is important. If Execute access is not available for a directory the Write access will have no meaning (you must be able to get to a directory before you can modify its contents). It is also possible to type the contents of a file without being able to list it in a directory! If Read access is not available for a directory, you may not list the files located there. Yet, if you have Execute access to that very same directory and know the exact name of the file you desire, you may display its contents with the dump, display, or type commands (try this out!).

As listed before, here in our discussion on groups, access privileges are kept for three classes of people. The classes are the **OWNER** of the file, the **GROUP** that owns the file, and the general **PUBLIC.** Remember, a user falling into more than one class will have the access privileges of all the classes to which they belong. To view the access privileges of a file, use the list command with the –e or –l option. Remember to use the –d option if you wish information on a a directory itself and not its contents. The classes of access privileges are listed in the order presented above.

SYSTEM PROGRAMMERS CORNER

For those of you who dabble in assembly programming and operating system details, the System Programmers Corner will try to keep you informed and offer you a forum to discuss your findings. This section, however, will only offer operating system type features and hints, versus the latest sorting algorithm. If you have any ideas to contribute to the SPC, please forward them to SOFT TIPS, in care of I/O News.

Cromemco's CROMIX manual many times alludes to the ability to format disks without setting aside an area for the boot tracks. This would be very handy when we know the disk is to be used only for data and we would like to gain whatever extra space could be made available when formatting. Unfortunately, the ability to not allocate a boot area is not a feature of the formatting process, but rather of CROMIX's "make file system" command. This is of no help to CDOS and C-10 users but may prove interesting anyway.

The disk is split up into four logical parts by Cromemco (with CROMIX titles in parenthesis): **disk parameter area** (identifier and super-block), **boot area, pointer area** (inode table), and **data area.** The parameter area is of fixed size under CROMIX, and our goal is to allocate all we can to the data area. To do this, we want to minimize the boot and pointer areas as much as possible. MAKFS has options, some currently undocumented by Cromemco, which give the user this capability.

The documented option, –i, allows you to specify the number of inodes to allocate for the disk. One inode is required by each file, each directory, and each device created on the disk. Any temporary or backup files created by your utilities will also require an inode. Do not be too stingy with inodes, it is important to allocate as much as will be needed. Our experience with disks has always been that a system crash occurs when you run out of inodes. For calculating storage purposes, four inodes make up one block.

To decrease the size allocated for a boot track, which is not needed since we are making a data disk, we will use the undocumented option, –b. This option allows you to declare in which block the inode table should start. When this option is not used, the system defaults to start the inode table in the 20th block on the disk (the blocks numbered 2 through 19 are set aside for a boot area). With this option, you can specify a starting block number as low as 3 to free up 17 additional blocks (though the starting block number can logically be as low as 2, the program does not allow it!).

This information is quite handy for 68000 CROMIX users who are implementing a RAM disk. Without this method, a 64K RAM disk will only have 50K free after initialization. With this method, RAM disks have been designed with up to 60K of free area.

One final point, if Cromemco used this method when they shipped their software, many of their packages which are sent out on multiple disks could actually go out on fewer diskettes (WE TRIED IT!!!). FORTRAN-77, for example, could be placed on one small floppy instead of two if the –b and –i options had been used on their diskettes.

MORE ON OPERATING SYSTEM UPDATES

As mentioned last time in Soft Tips, SUDS subscribers should have recently received updates to their operating

systems. Well, one group has so far been left out, mainly due to hardware shortages. I am referring to C-10 users who should expect to receive their SUDS update to Release 5 within the next two months. It is important to note, this update will include not only disk software but a new ROM chip which is to replace one inside your C-10. Make sure you have SUDS for every C-10 you own, or you will not receive enough of the chips and the release will not help your machines.

Release 5 will provide C-10 users with many enhancements, most importantly an operating system located in ROM! The advantages of a ROM based operating system are: no loading from disk on boot, and larger user memory area (or TPA as CP/M calls it). All screen and disk operations have been modified to provide increased speed, and the system will now support additional disk drives and printers. More complete CP/M compatibility is also offered, including implementation of call 31 (see "input" in the preceding I/O News). All Cromemco followers should be pleased to see this machine maturing so well.

As was mentioned last time, all CROMIX SUDS subscribers also recently received updates. A lot of talk has been generated about one of the new utilities, **FLUSH.** Though there seems to be no dispute as to the effectiveness of the utility, two questions/comments always seem to arise: The constant usage of additional memory and the delay caused when the system flushes its buffers, especially with a lot of users who have a lot of files open. To avoid the speed loss during flushing, it is important to specify the proper number of seconds the system should pause in between each flush of the system. Where power problems are not common, pause times of 300 seconds or more are preferred.

If system memory was more your concern than system speed, we will show you how to create your own FLUSH program. This program will flush the disk drive buffers and then exit; it will not loop continuously like the standard FLUSH program. Therefore, this utility can be called at any time by a user who wishes to make sure his most recent data changes are recorded onto the disk. Another application of this utility can be seen in the new LOGOFF command file below. To generate your own FLUSH utility, input the following commands as a privileged user:

```
# d /bin
# rename flush.bin flush.ori
# create flush.bin
# access rewa.e.e flush.bin
# chowner bin flush.bin
# patch flush.bin
)s
000000: 00 cf,52,38,05,21,00,00,cf
000008: 00 46,06,02,cf,1c,21,ff,ff
000010: 00 cf,46
000012: 00 .
)e
#
```

HANDY COMMAND FILES

We will now visit an old friend among command files, **logoff**, and a new command file **.logoff**, to work with our .startup.cmd file from our last column. The logoff command will not be changed for the most part, except to interface it with the new .logoff.

We use the the IF command to determine if a .logoff.cmd file exists in the home directory of the user. If the personal .logoff file exists, it is immediately executed. Then the rest of the system logoff file is executed as before. Then the rest of the system logoff file is executed as before. The personal .logoff file can be used to reverse any modifications the user made with a personal .startup file, or can be used to type reminders of events, etc., just like the personal .startup file. These command files follow.

Name: LOGOFF

Purpose: Controlled exit from the system offering more than the standard CROMIX login prompt.

Setup: None required

Listing:

```
if -r ../.logoff.cmd sh -q ../.logoff
/bin/flush
/bin/echo -n '^[*'
/bin/time
ty /etc/startup.msg
kill -3 0
```

Notes:

The above listing is only an example. Any commands desired can be put in between the first and last line. The first line looks for the personal .logoff file and executes it if it can be found. Our next line runs our modified flush utility (see above) to make sure all of the user's data has been removed from its buffers and written to disk. Make sure you do not use Cromemco's FLUSH in this way, as its looping feature will not let you logoff. Our example then clears the screen (here on a Televideo 925 terminal). The TIME utility is then called to show the system date and time at the top of the screen. The standard start-up message is then displayed by the fifth line; this is the same message displayed upon system boot. The final line will abort all current processes of the user, except those set running in background, and log him off the system. In other words, this line will issue the standard CROMIX login prompt. REMEMBER: All lines other than the first and last lines are shown here as an example only, you may place anything here that pertains to your situation.

Warning: This procedure proves fatal if run by a privileged user from a terminal that had originated the CROMIX Spooler. In that situation, a privileged user should simply use 'ex' to log off the system.

Example:

line will issue the standard CROMIX login prompt. REMEMBER: All lines other than the first and last lines are shown here as an example only, you may place anything here that pertains to your situation.

Warning: This procedure proves fatal if run by a privileged user from a terminal that had originated the CROMIX Spooler. In that situation, a privileged user should simply use 'ex' to log off the system.

Example:

```
% logoff
Wednesday, May 9, 1984                    18:04:00

For information about CROMIX, log in as "newuser".
Any user may access this same information by typing
"newuser" once logged in.

68000 CROMIX Operating System version 20.56
Copyright © 1983 Cromemco, Inc.

Login:
```

Name: .LOGOFF

Purpose: Executes a preset list of commands every time a user logs off the system with the command logoff. The following is an example only, any commands could be placed in this file.

Setup: None required

Listing:

```
/bin/mode im
/bin/echo -n '^[>'
```

Notes:

This command file is designed to be the opposite of the .startup.cmd in the last issue. First, the immediate echo mode of the terminal is then turned back on, returning it to this system's standard state of immediate echo mode on. Finally, the special key click feature of the terminal (here a Televideo 925) is returned to its normal state of off.

Example:

(((SAME AS ABOVE)))

Use This Handy Application to Become a Member of The I.A.C.U. and have I/O News Sent Directly to You.

Application for Membership

Please start my Membership in the International Association of Cromemco Users right away. I have enclosed my
( ) Check ( ) Money Order in the amount of

$_____ for a ( ) year membership.
(U.S. Dollars only please)

☐ International applicants: add $10.00 (U.S.) if you prefer to pay through your local bank.

Membership Rates in the United States:
( ) 1 yr. = $35.00 ( ) 2 yr. = $65.00
( ) 3 yr. = $90.00

Membership Rates in Canada and Mexico:
( ) 1 yr. = $41.00 ( ) 2 yr. = $77.00
( ) 3 yr. = $108.00

Membership Rates in all other countries:
( ) 1 yr. = $48.00 ( ) 2 yr. = $81.00
( ) 3 yr. = $129.00

Name: _____

Title: _____

Company: _____

Mailing Address: _____

_____

City: _____ State: _____

Country: _____ Zip/Postal Code: _____

Phone: _____ _____ _____
Area code number extension

Telex: _____

Now Available in the U.S. and most other countries on your VISA or Mastercard
(U.S. Dollars drawn on U.S. banks only **please**. Please Type or Print Clearly)

( ) VISA/Mastercard _____
(Expiration Date)

_____
(Your Full VISA or Mastercard Number)

Date _____ Signature _____

_____
(Name exactly as it appears on card)

The International Association of Cromemco Users is designed to provide its Members with the information they want. Help us deliver by answering the following questions. You may check more than one block as applicable: My field is:

( ) Professional Services ( ) Engineer: _____
(Accounting, Dentist, Law Medicine, Other _____) (please specify)
( ) Home Use
( ) Wholesaler or Distributor ( ) Retail Business
( ) Educational Institution ( ) Government:

_____ _____
(indicate level) (indicate branch)

My Primary Uses are:
( ) Business
( ) Business & Personal
( ) Educational
( ) OEM
( ) Personal Only
( ) Process Control
( ) Other: _____
( ) _____
( ) _____

I Want to Know More About the following Packages:
( ) Accounting
( ) Educational (adult)
( ) Educational (child)
( ) Computer Graphics
( ) Inventory Control
( ) Investments
( ) Home Economics
( ) Medical Research
( ) Process Control
( ) Sports & Games
( ) Other: _____
( ) _____
( ) _____

Describe Your Present System: (use brands and model numbers)
Computer: _____
Memory: _____
I/O: _____
Disk: _____
Terminals: _____
Printers: _____
Other: _____
_____

List Software Now in Use: (Packages, Special Operating Systems, etc.)
_____
_____
_____
_____

What Types of Software Would You Like to know more about?
_____
_____
_____

Would you be interested in preparing an article of interest to members?
☐ Yes
Subject Matter: _____
_____
_____

☐ I've included an additional $8.95 for my reference binder (including first class postage in U.S., Canada & Mexico. International purchasers, please add $2.00 [U.S.] extra for postage.)
☐ Please send information pertaining to my exclusive IACU group insurance privileges (U.S. only)
☐ Reserve a **Periodic Software Applications Guide** for me at the Members-only pre-publication price (to be announced prior to publication).

Mail Your Membership Application to:
The International Association of Cromemco Users
P.O. Box 17658, Irvine, CA 92713 U.S.A.
For more information, call (714) 955-0432

6 To 1350 MEGABYTES

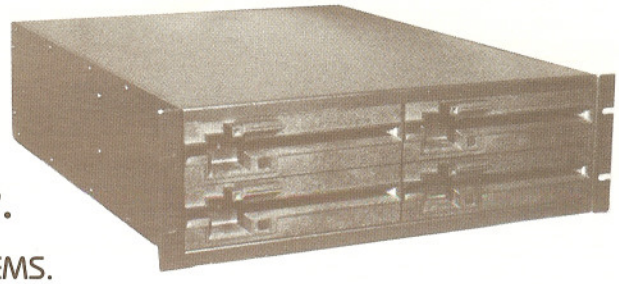
As low as $50.00 per Megabyte

HIGH PERFORMANCE HIGH CAPACITY-FLOPPY & HARD DISK SUBSYSTEMS

From

SYSTEMS ATLANTA, INC.

Over 400 models and configurations—custom cables
one year warranty, high speed backup utility supplied with each unit
Let us know your problems—We already have the solutions



SYSTEMS ATLANTA, INC.
Highway 5, P.O. Box 99
LEBANON, GEORGIA 30146
(404) 928-0240

EXPERTS IN CROMEMCO SYSTEMS SINCE 1977.

Discounts for authorized CROMEMCO dealers and OEMS.

Commercial Member Listing

Special Memberships are open to authorized Dealers and OEMs only. These memberships cost $350 per year, and entitle the member to a special listing on the Association's Referral Service Data Base, as well as this printed listing.

North America

Western United States

ACCOUNTABILITY SYSTEMS
700 South Tustin Avenue, Suite B
Orange, CA 92667
(714) 639-4570

An exclusive Cromemco dealership, Accountability Systems caters to the growing business and industrial base in Southern California. The Orange office supports the new personal computer system. Classroom training is available at both locations. CROMIX and Communication specialists. Developers of a professional medical billing package that can be used in single or multi-medical offices. The package provides full accounting for the medical office including monthly Patient Statements, Medicare & Medi-Cal Forms and Standard Insurance. Complete Business Accounting software that is customizable.

Orange Office:
Key Personnel: Michael L. Peterson, Systems Analyst
Kathleen Peterson, Office Manager
Pat McGuire, Jr., Software Systems
Bruce Hughes, CPA, Acctg. Consultant

AMERICAN COMPUTERS & ENGINEERS, INC.
Corporate Office: 2001 So. Barrington, Suite 204
Los Angeles, CA 90025
(213) 477-6751, Telex: 210-342-6365

Consulting Engineers. Full service company with complete line of hardware and software. Provides to end user on-site or in-house service agreements, fast turnaround on all repairs. Specializing in Structural Engineering, Drafting, Accounting & Word Processing software programs.

Key Personnel: Ghassan Dib, President, (Ph.D.Struct. Eng.)
Aziz Al-Khal, Vice President Sales, (Indust.Eng.)
Brian Gordon, Software Engineer

BRANCH OFFICES:

Orange County, CA: (714) 851-8700

Berkeley, CA: 2855 Telegraph, Suite 210
Berkeley, CA 94705 (415) 849-0177
Key Personnel: Kathy Kolder, Manager

Seattle, WA: 2001 6th Ave., Suite 1612
Seattle, WA 98121 (206) 583-0130
Key Personnel: Bob McNeel, Manager

Spokane, WA: North 9 Post, Suite 241
Spokane, WA 99201 (509) 838-9800
Key Personnel: Jay Townsend, Manager

Paris, FRANCE: 6 Rue Rochambeau 75009
Paris, France Ph: 285-4640
Key Personnel: Maurice Gaspard, Manager

Glasgow, SCOTLAND: 241-A W. George Street
Glasgow, Scotland PH: 248-2427
Key Personnel: Harvey Johnston

Athens, GREECE: 29 Arcadias Street
Athens, 608 Greece PH: 779-8868
Key Personnel: Dennis Ioakim

Major Market Area: Sales and Service, Worldwide

EXCALIBUR COMPUTERS
4558 Auburn Blvd., Suite 191
Sacramento, CA 95841
(916) 972-9252

Complete Systems house providing Sales, Service, Warranty Repair and Support for Cromemco Products. Custom Software developed in-house. Training available for CDOS, Cromix and Languages, as well as hardware. Has developed a Medical Billing Package and an Attorney Billing Package written in 32K Structured Basic. Market Cromix Drivers to implement concurrency on various different terminals.

Key Personnel: Robert Brown, Sales and Marketing
Curt Johnson, Systems Engineer
Jon Aimone, Software Support
Charles Stevenson, Design Engineer
Daniel Brown, Customer Support

Major Market Area: Sacramento, extending into Northern California

MCM ENTERPRISES
215 Hamilton Ave.
Palo Alto, CA 94301
(415) 327-8080
(415) 493-3333

Sales, Service, Integration, Installation, and Innovation — these are key words to describe MCM Enterprises. MCM is a full service computer solutions company with consulting, equipment, software, training, and service. MCM carries a full line of Cromemco Systems, NEC, Epson & Okidata Printers, Lear Tristar, Peachtree, Micropro (the WordStar People), ProCall Communications Software, and other specialized software. MCM Enterprises also offers full service on NEC Spinwriters, PerSci floppy drives, and all Cromemco equipment. MCM offers a variety of equipment and program service agreements. MCM also custom configures systems for international power requirements and has full export services. Call for training on CDOS, Cromix and languages, as well as hardware.

Key Personnel: M.C. Merchant (MSEE), Owner
T. Bidstrup, Sales
M. Ridgway (MS), Program Service
R. Blaylock, Equipment Service
M. Nadaire (MSEE), Mgr. Paris Office

Major Market Area:
Sales: San Francisco Peninsula & Nevada extending internationally.
Service: S.F. Peninsula and Nevada extending into N. California.
Paris Office: 4 Rue Paul Bert
92150 Suresnes, France
Tel (1) 506 33 03 TLX 610994F

MULTI-MEDIA VIDEO INC.
3350 Scott Blvd., Bldg. 21
Santa Clara, Ca. 95051
Tel: (408) 727-1733
Tlx: 171-577 MMV USA

Multi-Media Video, (MMV), markets bilingual Arabic/English Cromemco systems and peripherals throughout the Middle East. Installations have been made in the government and banking sectors; a complete Arabic banking system was developed for the latter.

Key Personnel: Ursula Burger, President
Michael Laugesen, Sales Manager
Clarissa Clayton, Marketing Manager

Major Market Area: Authorized dealers in Egypt, Saudi Arabia, and Pakistan.

Mid United States

ASGARD COMPUTING EQUIPMENT, INC.
121 West 6th Street
Neillsville, WI 54456
(715) 743-3344

823 5th Street
Menominee, MI 49858
(906) 863-6733

Exclusive Cromemco dealer with long term Cromemco association. Office includes on staff Engineers, Accountants & Chemists.

Key Personnel: James L. Bailey, President
Jerry Hagen, Vice President/P.E.
Ed Baetke, Secretary/Treasurer/Chemist

Major Marketing Area: Upper peninsula in MI, Northern & Western WI, Eastern MI.

COMPUTER CROSSROADS OF AMERICA, INC.
6 Terrace Shopping Center
Richardson, Texas 75081
(214) 231-6108 Twx/Telex 4991118

We are a CROMEMCO MASTER DEALER engaged in DEALER and OEM sales, service and support. We are in the top twenty-five dealers in the U.S. We have a consulting staff comprised of specialists in hardware, software and applications engineering. We are presently engaged in sales from the hardware level (equipment and/or software delivered in an unupened box) through the complete systems level where we take full responsibility for the system hardware configuration, instruction and maintenance of a system. As our name implies THIS IS THE CROSSROADS WHERE IT ALL COMES TOGETHER.

Key Personnel: Ed Fearon, Sales & Support
Danney Jarman, Sales & Support
Mark Shepheard, Sales
Bill Carnahan, Support

Major Market Area: Sales & Service Worldwide

TRADEWIND SYSTEMS
Box 96, West Highway 54
Liberal, KS 67901
(316) 624-8111, IN KS 1-800-362-9000
Outside KS 1-800-835-2057

Exclusive Cromemco dealer, specializing in complete business systems. Provides consulting services. Full inventory.

Key Personnel: David Fuller, Store Manager
Ray Cole, System Development
Kevin Elmore, System Development
Clark D. Stewart, President
Wayne Stewart, Vice President

Major Market Area: Sales: S.W. Kansas, extending to Colorado, Kansas, Oklahoma, Texas, New Mexico.
Service: S.W. Kansas

SYNERGISTICS INTERNATIONAL LTD.
35 Fountain Square Plaza, Suite 207
Elgin, IL 60120
(312) 695-7775

Full inventory of Cromemco hardware and software. Custom software developed in-house. Vertical market packages available include: Chiropractic Clinics; Architectural Woodwork Job Costing; Social Service Agency Accounting; Auctioneering. Specializing in providing turnkey systems to small and medium sized businesses.

Key Personnel: Jim Knowles, Pres. (Sales)

Major Market Area: Sales: Chicago and suburbs, extending to entire U.S. and the U.K. Service: Chicago and suburbs.

Eastern United States

CCS, INC.
A Computer Services Company
733 Third Avenue
New York, NY 10017
(212) 986-7520

Large Cromemco OEM specializing in custom applications on Cromemco Hardware. Full range of services including hardware sales, rentals, long and short term leasing, custom programming and continuous hardware and software support. Specialists in database and large scale financial applications.

Key Personnel: Richard Levey, Vice President
John Ruffo, Vice President

Major Market Area: U.S. and Major cities throughout the world.

COMPUTER CLOSET INC.
45F Route 303
Valley Cottage, NY 10989
(914) 268-5161

Complete systems house providing sales, service and support for the full line of Cromemco hardware and software. Provides system planning and design for custom applications in business, education, and professional fields. Regular schedule of seminars and training classes offered.

Key Personnel: Rick Townsend, President

Major Market Area: Sales: Northeast U.S. and East Coast
Service: Continental U.S.

COMPUTER SYSTEM & TECHNOLOGY, INC.
226 Sherwood Avenue,
Farmingdale, NY 11735
(516) 420-1470

Cromemco dealer since 1980, we are known for our resourcefulness and good service. We specialize in 68000/UNIX computers, database, and graphic. Among the products we develop are a graphic workstation for CAD application, and the RETAIL CONTROLLER™ that connects the cash register to the computer for inventory control, sales analysis, and cash management. Key personnel have strong background in engineering, financial markets, and import/export trade.

Key Personnel: Mike Fung, President
Fanny Ho, Manager
Salina Ho, Project Leader

Major Market Area: New York, extending to entire U.S., Canada, China, Europe, and South America

CUSTOM COMPUTER SPECIALISTS, INC.
300 Vanderbilt Motor Parkway
Hauppauge, NY 11788
(516) 231-1155

Full service systems house with retail showroom. Full line of Cromemco hardware, software, accessories, and literature. Provides warranty service, diagnostics, consultation, systems analysis, and custom programming. Special management software for attorneys, mass transportation scheduling, reservations, delivery manifests, education, small business.

Key Personnel: Gregory G. Galdi, President
Linda M. Miller, General Manager

Major Market Area: Sales: Northeast U.S., extending to East Coast.

Service: East Coast extending to Continental U.S.

METRO COMPUTER
Division of F.S.L. Corp.
110 Harvard Street
Brookline, MA 02146
(617) 277-5115

A complete service and support dealership committed to a full line of Cromemco products. Services encompass custom designed hardware and software for individual interfacing needs such as communications applications, multi-vendor configurations, add-ons, and upgrades. All Cromemco systems on display and available for hands-on demonstration, including color graphics system. In addition to on-site warranty repairs, extended maintenance is offered on either an hourly basis or by contract.

A wide range of software and peripherals compatible with Cromemco systems is available. Metro Computer specializes in office productivity, accounting packages and specialty markets. The consulting division of F.S.L. Corporation emphasizes DESIGN AND ENGINEERING for the INDUSTRIAL and DEFENSE area. Our staff of engineers and technicians are standing by for customer support.

Other Lines: Apple, Hewlett Packard

Key Personnel: Frederick S. Lebow, C.E.O. (Engineer)
Dwight Calhoun, Director of Engineering
Robert J. Durante, Vice President of Finance
Melissa Lavers, Sales Manager
James Ritteman, Technical Support (Sales)

Primary Marketing Area: Massachusetts
Extended Marketing Area: New England States

SYSTEMS ATLANTA, INC.
P.O. Box 99
Highway 5, Toonigh Road
Lebanon, Georgia 30146
(404) 928-0240

As one of Cromemco's oldest dealers, Systems Atlanta is well experienced in hardware and software implementation. With over 1000 systems installed and a full staff of highly seasoned employees, Systems Atlanta offers technical support for operating systems, application software and hardware design. Specific configurations include telecommunications, graphics, data base management as well as fully integrated accounting systems. Systems Atlanta has authored several specific applications packages such as Manufacturing and Inventory Control, Church Management, Job Costing and Unix based programs.

Key Personnel: Charley Dobson, President & G.M.
Betty Dobson, Dir. of Finance & Admin.
Gary Kendrick, Dir. of Marketing

Steve Garrison, Operations Manager

Major Market Area: Worldwide, with exports to South America, Europe, the Middle East and Canada.

Canada

COMPUTER SOLUTIONS
1700 Varsity Estates Drive N.W.
Calgary, Alberta
Canada T3B 2W9
(403) 286-8459 Telex: 03-827506

Complete sales, service and support center for Cromemco and a variety of quality peripherals and support products. Specializing in support over long distances, OEM accounts, custom software, solutions to computing concerns.

Key Personnel: Bob Pyle, General Manager
Mark Dutchuk, Customer Support
John Shepherd, Sales Manager

Primary Marketing Area: Western Canada
Extended Marketing Area: Eastern Canada, Northern/North-Western U.S.

D.E. SYSTEMS LTD.
1284 Wellington St.
Ottawa, Ontario
Canada, K1Y 3A9
(613) 729-5164

D.E. Systems Ltd. is a full service company offering Cromemco Hardware, Software Development, Education and Application Programs. We have developed integrated Inventory, Point-of-Sale, Invoicing, Accounting and Sales Analysis programs as well as a Courier Package. We specialize in Cromemco Computers for government and small businesses. We have most Cromemco products in stock and offer technical support on the hardware and software. We offer maintenance of all Cromemco equipment and related peripherals.

Key Personnel: Bruno Dugas, President
Keith Corkum, Director (Systems Development)
Dwight Presley, Senior Analyst

Major Market Area: Eastern Canada

Mexico

SOPORTE ADMINISTRATIO COMPUTACIONAL, S.A.
15 de Mayo 1111 Pte.
Monterrey, N.L., Mexico
Tels. (83) 43-83-40 and 44-62-69

Complete line of Cromemco hardware and software in inventory. Specializing in Business with Software packages (in CROMEMCO Structured Basic) in Spanish according to Mexican Laws and Taxes including Accounting, Payroll, Accounts Receivable and Payable, Inventories, etc., provides full service facility, including technical consulting, custom software, warranty and repair.

Key Personnel: Juan Angel Perez, Director
Luis Ernesto Rodriguez G., Marketing
Delfino Juarez, MSEE Technical Support

Major Market Area: Northeast Mexico

International

Europe

AGRO MARKETING
B Adzije 7/1, 41000 Zagreb
Yugoslavia
41 417-662 Telex: 862-21741 YU AM 2G

Large full-service facility, with complete line of Cromemco products and proprietary software. Specializing in software development, interfacing, and special medical computerized equipment.

Key Personnel: T. Raguz, Director (Marketing)
N. Ivancic, Software Manager
B. Krtolica, Customer Support (Hardware)

Major Market Area:
Sales & Service: Internationally, primarily Yugoslavia

Middle East

MICRO COMPUTER SYSTEMS MARKETING CENTER
P.O. Box 1446
Jeddah, Saudi Arabia
(966) (2) 651-7707 or 653-0580/Telex 928-403068 MICSYS SJ

Authorized Dealer in Jeddah-Saudi Arabia (Western Zone) for Cromemco sales and maintenance of computers, peripherals, software development and design. Strong Arabic Software development.

Key Personnel: Abdul Rahman H. Attar, General Manager
Issam Al Safadi, Administrative Manager
M. Ali Khan, Marketing Executive

REALTIME ENGINEERING & DATA ANALYSTS
P.O. Box 278
Dhahran Int'l Airport
Dhahran, Saudi Arabia
(966) (3) 8649043/Telex: 928-670480 READAK SJ

P.O. Box 6156
Jeddah
Saudi Arabia
(966) (2) 6531502

Sales and maintenance of computers, peripherals and supplies within the areas of automation, industrial, business and office. Security systems. Strong in developing ARABIC SYSTEMS (hardware and software) and turnkey projects. Large simulators and facsimile.

Key Personnel: A.A. Salamah, Administrative Director
Nasir Jamil, Manager Digital Systems Div.
Ziyad Ismail, Software Design and Development

Major Market Area: CROMEMCO distributor for Middle East (Saudi Arabia, Gulf Emirates, Iraq, Syria, Jordan, Lebanon)

Far East

ASAHI GLASS
Electronics Group
Special Products Marketing Div.
1-2 Marunouchi, 2 Chome
Chiyodaku, Tokyo 100
Japan
781-24616/Telex: 24616 ASAGLAS

Complete line of Cromemco hardware and software in inventory. 700 sq. foot training room. Specializing in O.S. modifications. Full service facility, providing technical consulting as well as warranty repair service.

Key Personnel: Shigeo Satoh, General Manager (systems)
Norimasa Hori, Manager (sales)
Shinichi Watanabe, Tech/software

Major Market Area: Japan

CHINA DATA PROCESSING CENTER
6th Fl., Yu Ming Mansion
7 Roosevelt Road, Section 1
Taipei, Taiwan
Republic of China
02-392-2284/Telex: 785-19844 CDPC CHOWE

One of the largest importer/exporters of computer business/industrial control systems in Taiwan. With several years of computer engineering experience, Tien Sheng provides turnkey basis and reputable service.

Key Personnel: Mr. R. Sheu
Mr. C.K. Cheng
Mr. M.S. Hu

Major Market Area: Taiwan, Republic of China

SUPER-NATURE COMPUTER CO., LTD.
4F-1, No. 239, Ta-An Road, Sec. 1
Taipei, Taiwan. Republic of China.
(02) 705-2442, (02) 700-4858/TELEX: 13937 SNCOMPUT

Our Company primarily sells CROMEMCO computer equipment with high technology and experience, providing installation, warranty repair service and customer education.
Specializing in O.S. modifications, software and hardware development for business and industrial applications.

Key Personnel: Miss Shu-Chin Kuo, President
Mr. Mark Yeh, Sales manager
Mr. Morgan Chen, Import/Export department
Mr. Ringol Shiung, Chief of R&D department

Local Cromemco User's Groups

Arizona Association of Cromemco Users
Contact: Jo Ann Drake, President
2207 West Eugie Avenue
Phoenix, AZ 85029
(602) 993-9589

Bay Area Cromemco Users & Programmers (BACUP)
Contact: Raymond Barglow or Alan Walworth
United Word & Data Processing
2345 Fulton Street
Berkeley, CA 94704
(415) 841-0708 or (415) 548-2692

Cromemcohorts
Contact: Dr. Brent Lowensohn
4747 Sunset Blvd.
Los Angeles, CA 90027
(213) 667-8972

Cromemco Users' Group Holland (CUGH)
Contact: Joop Kohler, Secretary
P.O. Box 120
2910 AC Nieuwerkerk a/d IJssel
The Netherlands 01803 - 3300

Cromemco Users' Group
Contact: Peter Norman
The University of Newcastle Upon Tyne
Department of Chemical Engineering
Merz Court, Claremont Road
Newcastle Upon Tyne NE1 7RU
England
Newcastle 28511, Ext. 3278
*Publishes Cromemco Users' Newsletter (CUG)

Cromemco Users' Group Ontario, Canada
Contact: Lloyd Parker
Hiram Walker Resources Ltd.
Suite 600
1 First Canadian Place
Toronto, Ontario
Canada M5X 1A9
(416) 864-3349

Cromemco Users of Orange County, California
Contact: Michael Peterson
Accountability Systems
700 South Tustin Avenue
Suite B
Orange, CA 92667
(714) 639-4570
Meets third Tuesday Monthly

Insystems Pty. Ltd.*
Contact: Norman Rosenbaum
337 Moray Street
South Melbourne, Victoria
3205 Australia
(03) 690-2899, telex AA30458
*Publishes "Cromemco UPDATE
a bi-monthly newsletter

Illinois Users' Group
Contact: Jim Knowles
P.O. Box 631
Elgin, IL 60120
(312) 695-7775

Indonesian Cromemco Users' Group (ICUG)*
Contact: Zafir M.A. Pontoh
Computation Lab
Department of Regional & City Planning
Bandung Institute of Technology
10 Ganesha
Bandung, Indonesia
(022) 82051 ext. 360
*Publishes "BERKALA ICUG."
a monthly newsletter

Microcomputer Users' Group
Contact: Jim Lenz
1165 Barbara Drive
Cherry Hill, NJ 08003
(609) 428-6701

Northwest Association of Cromemco Users (NWACU)
Contact: Jim Illman
403 S. Brandon
Seattle, WA 98108
(206) 763-2099

North Texas Cromemco Commercial Users' Group
Contact: Jerrell Johnson
1131 Winterwood
Lewisville, TX 75067
(214) 221-1437
Or call Rocky Hall
@ (214) 398-1595
Meets first Wednesday bi-monthly

NY, NY Users' Group
Contact: Charles Perrella
45F Route 303
Valley Cottage, NY 10989
(914) 268-5137

SaCromemco Users
Contact: Alan Whitman
Box 244
Rancho Cordova, CA 95670
(916) 635-6070

Silicon Valley Cromemco Users
Contact: Alan O'Neill
(415) 969-3854 or Emily Ott (415)
854-5818
Meeting place provided by:
MCM Enterprises
215 Hamilton Avenue
Palo Alto, CA 94301
Meets second Tuesday monthly

W.A. Cromemco Users' Group
Contact: Rae Canning
c/o The W.A. School of Computing
2/294, Rokeby Road
Subiaco, Western Australia 6008

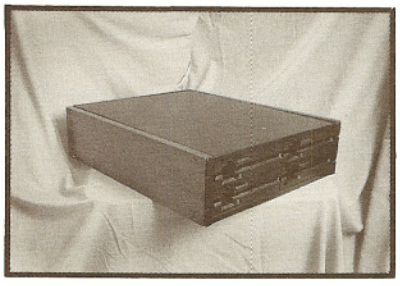
West Germany Users' Group
Contact: Glynnis Long
Tesco GmbH
P.O. Box 10
8714 Weisentheid
West Germany
09383-1237
Total fluency in English & German

DISK DRIVE SUBSYSTEMS

THE SOLUTION TO DISK STORAGE PROBLEMS

8" Subsystem Matches System One

CDC Phoenix, Lark and Wren Subsystems
(not shown)
30 MB TO 96 MB
FROM $1,200[00]



8" Combination Pack

CP/M[1] 2.2 for C-10

Put our CP/M 2.2 on your C-10 and see the difference in performance.

HARD DISKS for C-10 are available NOW!

ProCALL Communications Software for CDOS[2], CROMIX[3] and CP/M available.

1 - 800-346-1750

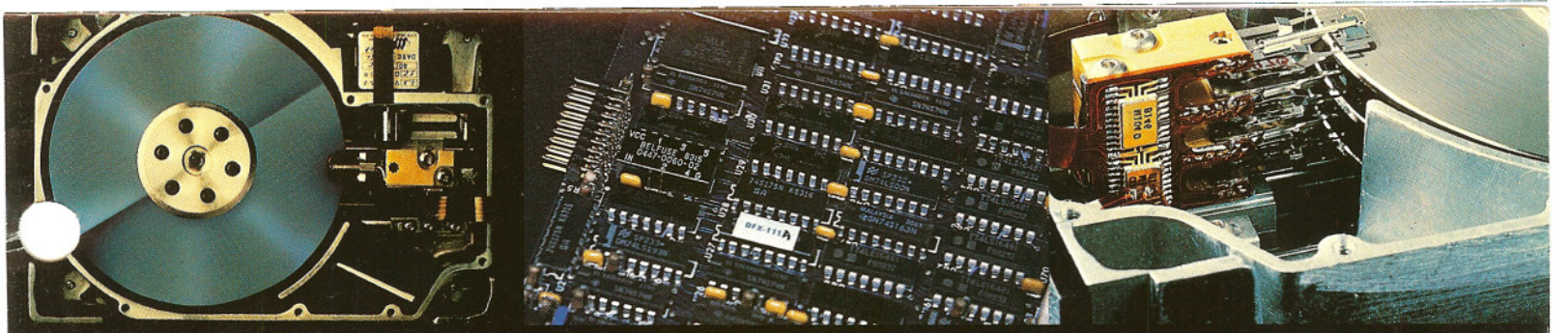
"Providing Enhancements to Cromemco Systems since 1977"

1 - 800-346-1750
Telex: 49904013

For information, please contact:

Royal Data, Inc.
2199 Garden Street
P.O. Box 2745
Titusville, FL 32780
(305) 267-1960

Trademarks Digital Research, [2]Cromemco, [3]Cromemco



How do you spell relief?

AUXIN™ The ultimate mass storage relief machine for your Cromemco computer.

With up to 18 megabytes (300 Mb on EXOCET™ models) of hard disk capacity, you can handle extremely large data bases. Simply access, process, and update all that data in micro-seconds.

Which, after all, is what serious personal/business computing is about.

PERFORMANCE.

AUXIN series hard disks are truly beautiful engineering marvels.

FAST.

Data moves swiftly in and out at 5 million bits per second.

QUIET.

Even with the disks spinning at 3600 rpm inside.

SOLID.

Which is why AUXIN series drives carry a warranty few others would dare offer. 2 years on disk drive, 12 months on controller and chassis, including parts and labor.

EASE OF USE.

Just plug the AUXIN drive controller into your System One, System Two, System Three, or even C-10, and you are ready to go.

AUXIN drives come with plain English instructions and the software required to make them perform marvels with CROMIX or CDOS. CP/M Plus will be available early 1984.

LUXURY AT ECONOMY PRICE.

The best way to order an AUXIN series drive is right from your local dealer, or directly from the factory. You pay considerably less than you would for a lesser system sold elsewhere. Powerful standalone AUXIN mass storage systems start at $2495.

AUXIN 18c-18 Mb Hard Disk for C-10	$2,495
AUXIN 18s-18 Mb Hard Disk for CS1, CS2, CS3,	$2,495
AUXIN 18a-18 Mb Hard Disk Add-on	$1,995
EXOCET 315-300 Mb Hard Disk for CROMIX Systems	$12,995

Call or write for information on other products, models and/or special configurations available.

To reach our professional sales staff, dial (415) 366-4610.

You may charge the AUXIN to your VISA or MasterCard or you may send a check, money order or wire transfer. Your AUXIN mass storage system will arrive in virtually days.

And your Cromemco will get a much needed boost in mass storage horsepower.

RELIEF.

Auxin's high performance, low cost, extended warranty, and immediate availability.

Dealer Inquiries Invited

National Cryptographic Devices, Inc.
Peripheral Division
635 Bair Island, P.O. Box 2938
Redwood City, CA. 94064
(415) 366-4610

68000-based systems. Just tell us what you need.

68000-based systems to fit your application.

Right from the pages of our catalog, we can deliver 68000-based supermicro systems to match virtually any application.

Including yours.

Here's how.

Built on the IEEE-696 (S-100) bus, Cromemco systems offer up to 21 board slots. And a family of 35 boards — CPU, memory and specialized I/O — to fill the slots any way you choose.

At the heart of each system is our 68000/Z-80 dual processor. Backed by as much as 16 Mb of error-correcting RAM. Full multi-tasking capability. I/O to handle up to 16 terminals.

And that's just the beginning.

You can select single or dual floppies, 5¼" or 8". A 21 Mb 5¼" Winchester hard disk. And a nine-track tape drive.

We can accommodate your taste for the exotic, too. With boards like our SMD interface that supports up to 1200 Mb of disk storage. An NTSC standard color graphics interface. A TV camera digitizer. A/D and D/A converters. An IEEE-488 bus interface. Communications. And more.

Intelligent workstations.

Then, if you're designing a distributed processing system, you'll want to take a look at our C-10 personal computer. The Z-80-based C-10 can serve our 68000-based systems as a powerful intelligent workstation in a distributed processing mode. Or as an independent personal computer with its own floppy storage.

High-level languages and applications software.

That brings us to software. It starts with CROMIX®, our UNIX™-like operating system that you're free to tailor to your application.

CROMIX can execute both 68000- and Z-80-based programs. So right along with your 68000-based packages, your system will accommodate a wide selection of CP/M® software written for the Z-80.

And our high-level language support is second to none. From a 68000 Macro Assembler. To 68000 FORTRAN 77, PASCAL, GSA-certified high-level COBOL, C and BASIC.

Cromemco means business. Your business.

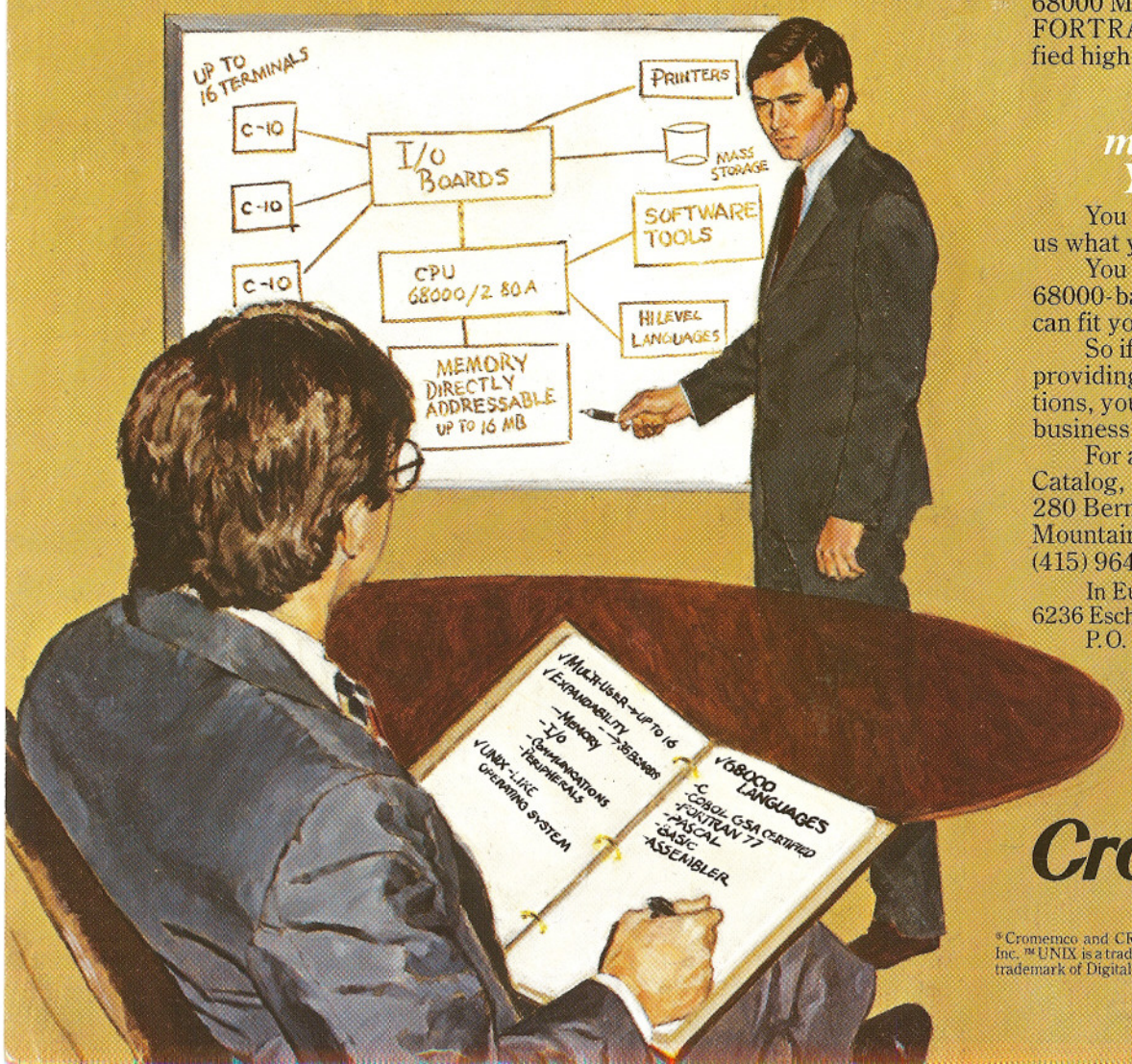
You see, when we say, "Just tell us what you need," we're not kidding.

You won't find another family of 68000-based microcomputers that can fit your needs as exactly as ours.

So if you're in the business of providing specialized computing solutions, you really should be doing business with Cromemco.

For a copy of our Systems Catalog, contact: Cromemco, Inc., 280 Bernardo Avenue, P.O. Box 7400, Mountain View, CA 94039. (415) 964-7400.

In Europe: Cromemco/GmbH, 6236 Eschborne 1, Frankfurter Str. 33-35, P.O. 5267, Frankfurt Main, Germany or Cromemco Ltd., The Cambridge House, 178-182 Upper Richmond Rd., Putney, London SW15 England.



Cromemco®